Generic Methodology in Report Building by Using SAS® Macros

Irene Zhao, Allergan Inc., Irvine, California

ABSTRACT

SAS programmers in the pharmaceutical industry are frequently under time constraints in generating statistical tables for clinical reports. Most tables are similar in format. If we create a generic table generation program, we can greatly reduce the statistical table generating time. This paper discusses how to create generic statistical tables generating programs for clinical reports by using SAS macro techniques and the REPORT procedure (PROC REPORT). There are two major macro structures used in the provided example, a demographic table program. Each macro determines descriptive and/or inferential statistics for continuous variables and categorical variables. The statistical analysis for the categorical variables includes sample sizes, percentages and probability values (p-value) generated by the FREQ procedure, while the analysis for the continuous variables includes means, standard deviations (STD), ranges and p-values generated by the MEANS procedure and the GLM procedure. This example shows a generic way to build demographic tables, and the method can be applied to other tables that display continuous variables and/or categorical variables.

INTRODUCTION

Many programmers are revising old programs to fit new requests. Some of the programs are easy to revise, and some could lead one into a maze of disaster. A well structured program can make the revision smooth, easy, and time saving. This paper introduces a method to compose a program in a generic structure for report building. There are four steps in this table generation method:
- Prepare the data: Read in data from one or more data sets and create new variables when needed.
- Write the macros which perform the analysis requested.
- Make proper macro calls.
- Output the table by PROC REPORT.

As an example, demographic table building is introduced. A demographic table is requested and its template is as in Chart I.

Chart I:

Table 1 Patient Demographics: Age, Sex, Race, and Height			
Variable		**Study Drug**	**Vehicle**	**P-value[a]**
Age (Years)	N	XX	XX	X.XXX
	Mean	XX.X	XX.X	
	STD	X.XX	X.XX	
	Min	XX	XX	
	Max	XX	XX	
Sex	Male	XX(XX.X%)	XX(XX.X%)	X.XXX
	Female	XX(XX.X%)	XX(XX.X%)	
Race	Caucasian	XX(XX.X%)	XX(XX.X%)	
	Black	XX(XX.X%)	XX(XX.X%)	
	Other	XX(XX.X%)	XX(XX.X%)	
	Caucasian	XX(XX.X%)	XX(XX.X%)	X.XXX
	Non-caucasian	XX(XX.X%)	XX(XX.X%)	
Height (cm)	N	XX	XX	X.XXX
	Mean	XX.X	XX.X	
	STD	X.XX	X.XX	
	Min	XX	XX	
	Max	XX	XX	

[a] P-values for between-treatment comparisons. Age and height were analyzed by general linear models. Sex and race (Caucasian vs. Non-caucasian) were analyzed by chi-square.

DATA PREPARATION

There are four variables to be analyzed and presented in the request: age, sex, race, and height. Two of them are categorical (sex and race) and two are continuous (age and height). If the four variables are in the data sets read in, only one additional race variable (Caucasian vs. Non-Caucasian) needs to be created to calculate the p-value. In the example, a dummy data set is created by using the SAS/Basic DATA step CARD statement. For programming codes, see Appendix A, Part 1.

MACROS

There are two major macro pieces to be created for statistical analysis:

1. For continuous variables:
A variable analyzed to show its mean score is considered a continuous variable in this topic. Use PROC MEANS to output the descriptive statistics into a SAS data set, then transpose the data set into a treatment-column structure by PROC TRANSPOSE. Retrieve inferential-statistics' p-values from the statistics data set output by PROC GLM. Then, merge the two statistical data sets and store the merged data set into a new data set (named "NEW") by PROC APPEND.

2. For categorical variables:
A variable analyzed to show its response-frequency distribution is considered a categorical variable in this topic. Use PROC FREQ with option CHISQ to output descriptive and inferential statistics data sets. The output statistics include sample size, percentage, and p-values. Then, transpose the counts and percentages into a treatment-column structure by PROC TRANSPOSE. Perform the necessary manipulation on the p-values, such

as rounding the decimal digits, converting its type into character, and assigning a value of "N/A" if the p-value is not calculated, etc. Finally merge the counts and percentages with the p-values and store the merged data set into the "NEW" data set by PROC APPEND.

Since methods to calculate p-values vary, different macro subsets for alternate p-values, such as CHISQ, CMH, Fisher's 2x2 EXACT, Anova, Wilcoxon, etc., should be prepared for use. Only local variables should be used in these macro pieces to support report building generically. For programming codes, see Appendix A, Part 2.

PROPER MACRO CALLS

Make macro calls in the same sequence to be presented on the table. Macro calls are made in the example as follows:

```
%cont(age,Age (Years),1);
%catg(sex,Sex,sex.,chisq,2);
%catg(race,Race,race.,,3);
%catg(race2,Race,racea,chisq,3);
%cont(ht,Height,4);
```

After each macro call finishes running, the output records are stored in the "NEW" data set create by PROC APPEND. When all the macro calls finish running, the records in "NEW" data set are as listed in chart II.

OUTPUT TABLE

PROC REPORT is employed to format the table in the way requested. The table displayed in chart III is generated by the example program running under a VAX/VMS environment and by SAS System version 6.08. For programming codes, see Appendix A, Part 4.

Chart II:

```
                                       NEW.SSD

      OBS  _1              _2            CAT        ORDER  VARLAB        CATN  PVALUE

        1  10              10            N             1   Age (Years)    1    0.678
        2  40.3            46.7          Mean          1   Age (Years)    2
        3   8.26            9.94         STD           1   Age (Years)    3
        4  29              32            Min           1   Age (Years)    4
        5  51              62            Max           1   Age (Years)    5
        6   5 ( 50.0%)      5 ( 50.0%)   Male          2   Sex            1    1.000
        7   5 ( 50.0%)      5 ( 50.0%)   Female        2   Sex            2
        8   5 ( 50.0%)      5 ( 50.0%)   Caucasian     3   Race           1
        9   4 ( 40.0%)      4 ( 40.0%)   Black         3   Race           2
       10   1 ( 10.0%)      1 ( 10.0%)   Other         3   Race           3
       11                                              3   Race           3
       12   5 ( 50.0%)      5 ( 50.0%)   Caucasian     3   Race           1    1.000
       13   5 ( 50.0%)      5 ( 50.0%)   Non-caucasian 3   Race           2
       14  10              10            N             4   Height         1    0.613
       15  171.7           168.4         Mean          4   Height         2
       16   8.30            7.46         STD           4   Height         3
       17  158             158           Min           4   Height         4
       18  185             182           Max           4   Height         5
```

Chart III:

```
                                    Table 1

                    Patient Demographics: Age, Sex, Race, and Height

            Variable                  Study Drug      Vehicle       P-value[a]

            Age (Years)   N            10             10            0.678
                          Mean         40.3           46.7
                          STD           8.26           9.94
                          Min          29             32
                          Max          51             62

            Sex           Male          5 ( 50.0%)     5 ( 50.0%)   1.000
                          Female        5 ( 50.0%)     5 ( 50.0%)

            Race          Caucasian     5 ( 50.0%)     5 ( 50.0%)
                          Black         4 ( 40.0%)     4 ( 40.0%)
                          Other         1 ( 10.0%)     1 ( 10.0%)

                          Caucasian     5 ( 50.0%)     5 ( 50.0%)   1.000
                          Non-caucasian 5 ( 50.0%)     5 ( 50.0%)

            Height (cm)   N            10             10            0.613
                          Mean        171.7          168.4
                          STD           8.30           7.46
                          Min         158            158
                          Max         185            182

        [a] P-value for between-treatment comparisons. Age and height ware analyzed by general
            linear model. Sex and race (Caucasian vs. Non-caucasian) were analyzed by chi-square.
```

CONCLUSION

The programs, structured into the four steps described above, have clear logic flow and are easy to adopt from one project to another. Revising existed programs becomes smooth, easy, and time saving by changing the input data, macro calls and the report procedure (titles and footnotes, etc.). The macro pieces can be reused by other programs. Since each table has its own characteristics, it is necessary to modify macro pieces to fit the table. It would be convenient to set up a macro library storing a collection of different purpose macro pieces for use. This method can be used for many of the tables in an NDA submission and other similar reports.

APPENDIX A

```
options ls=88 ps=49 nodate nonumber mprint;

proc format;
    value sex 1='Male' 2='Female';
    value race 1='Caucasian' 2='Black' 3='Other';
    value racea 1='Caucasian' 2='Non-caucasian';
    run;

* Part 1:  Get demographic data ready;

data demo;
    input ptid $ site trt sex race age ht;
    if race=1 then race2=1;       * Caucasion;
```

```
    else if race^=. then race2=2;   * Non-cauc.;
    cards;
    101 1 1 2 1 29 175
    102 1 1 1 1 32 182
    103 1 2 2 2 42 158
    104 1 1 2 1 41 165
    105 1 2 2 2 40 165
    106 1 2 1 1 39 172
    107 1 1 1 2 49 174
    108 1 2 2 1 50 166
    109 1 1 2 2 51 158
    110 1 2 1 1 62 172
    111 2 1 1 3 29 175
    112 2 2 2 2 32 182
    113 2 1 2 1 42 185
    114 2 1 1 2 41 165
    115 2 2 1 2 40 165
    116 2 1 2 2 39 172
    117 2 2 1 1 49 174
    118 2 1 1 1 50 166
    119 2 2 1 1 51 158
    120 2 2 2 3 62 172
    ;

proc sort; by trt; run;

* Part 2:  Macro pieces for calculating statistics;

%macro cont(invar,label,order);

proc printto print="g&order..dat";
proc means data=demo noprint;
```

```
         by trt;                                          by order;
         var &invar;                                      if not first.order then pvalue=' ';
         output out=&invar n=nn mean=nmean                run;
                 std=nstd min=nmin max=nmax;
  proc printto; run;                              proc append base=new data=&invar force; run;


  data &invar(drop=nn nmean nstd nmin nmax);      %mend cont;
         length n mean std min max $12 varlab $12;
         set &invar;                              %macro catg(invar,label,fmt,method,order);
         order=&order;
         varlab="&label";                         proc printto print="g&order..dat";
         mean=put(nmean,5.1);                      proc freq data=demo noprint;
         std=put(nstd,6.2);                            tables trt/out=trt;
         array v{3} nn nmin nmax;                      tables trt*&invar/out=&invar;
         array c{3} $ n min max;                   proc printto; run;
         do i=1 to 3;
            c{i}=put(round(v{i},.1),3.);           data _null_;
         end;                                          set trt;
         run;                                          select (trt);
                                                           when (1) call symput("tot1", count);
  proc transpose data=&invar out=&invar.t;                 when (2) call symput("tot2", count);
         id trt;                                           otherwise;
         by order varlab;                              end;
         var n mean std min max;                       run;
         run;
                                                   data &invar;
  data &invar.t(drop=a b _name_ );                     set &invar;
    length _1 _2 $12 cat $13;                          order=&order;
    set &invar.t(rename=(_1=a _2=b));                  run;
    _1=a;
    _2=b;                                          proc sort data=&invar; by order trt; run;
    select (_name_);
    when ('N') do; cat='N'; catn=1; end;           data &invar
    when ('MEAN') do; cat='Mean'; catn=2;end;            (keep=cat order catn &invar stats varlab trt);
    when ('STD') do; cat='STD'; catn=3; end;             length cat $13 pct $5 stats $12 varlab $12;
    when ('MIN') do; cat='Min'; catn=4; end;             set &invar;
    when ('MAX') do; cat='Max'; catn=5; end;             by order;
    otherwise;                                           varlab="&label";
    end;                                                 catn=&invar;
    run;                                                 cat=put(&invar,&fmt.);
                                                         select (trt);
  proc sort data=&invar.t; by order catn; run;            when (1) percent=(count/&tot1)*100;
                                                           when (2) percent=(count/&tot2)*100;
  proc printto print="g&order..dat" new;                   otherwise;
  proc glm data=demo outstat=&invar.p;                 end;
         model &invar=trt|site/ss3;                    pct=put(round(percent,.1),5.1);
  proc printto; run;                                   stats=put(count,3.)||' ('||right(pct)|| '%)';
                                                       run;
  data &invar.p(keep=order pvalue);
         length pvalue $8;                         proc sort data=&invar;
         set &invar.p;                               by order varlab &invar cat catn;
         if _type_='SS3' and _source_='TRT';         run;
         order=&order;
         pvalue=put(round(prob,.001),8.3);         proc transpose data=&invar out=&invar.t;
         run;                                          id trt;
                                                       by order varlab &invar cat catn;
  data &invar;                                         var stats;
         merge &invar.t &invar.p;                      run;
```

-4-

```
%if %upcase(&method)=CHISQ %then %do;

proc printto print="g&order..dat";
proc freq data=demo noprint;
    tables trt*&invar/chisq;
    output out=&invar.p chisq;
proc printto; run;

data &invar.p(keep=order pvalue);
    set &invar.p;
    length pvalue $8;
    order=&order;
    pvalue=put(p_pchi,8.3);
    run;

data &invar(drop=&invar);
    merge &invar.t(drop=_name_) &invar.p;
    by order;
    if not first.order then pvalue=' ';
    run;

%end;
%else %if %upcase(&method)= %then %do;

data &invar(drop=&invar);
    length pvalue $8;
    set &invar.t(drop=_name_);
    by order;
    pvalue=' ';
    output;
    if last.order then do;
      cat=' '; _1=' '; _2=' '; pvalue=' ';
      output;
    end;
    run;

%end;

proc append base=new data=&invar force; run;

%mend catg;

* Part 3: Statistical analysis by macro calls;

%cont(age,Age (Years),1);
%catg(sex,Sex,sex.,chisq,2);
%catg(race,Race,race.,,3);
%catg(race2,Race,racea.,chisq,3);
%cont(ht,Height,4);

* Part 4: Output table;

proc report data=new headskip;
    column order varlab cat _1 _2 pvalue;
    break after order/skip;
    define order/order noprint;
    define varlab/order width=12 left 'Variable';
    define cat/display spacing=3 width=13 left ' ';
```

```
    define _1/display spacing=3 width=12 left 'Study
Drug';
    define _2/display spacing=3 width=12 left
'Vehicle';
    define pvalue/display spacing=4 width=10 left 'P-
value[a]';
    title 'Table 1';
    title3 'Patient Demographics: Age, Sex,Race, and
Height';
    compute after;
    line @1 88 *'_';
    line @1 '[a] P-value for between-treatment
comparisons. Age and height were analyzed by
general';
    line @1 '   linear model. Sex and race(Caucasian
vs. Non-caucasian) were analyzed by chi-square.';
    endcomp;
run;
```

ACKNOWLEDGMENTS

CONTACT

Irene Zhao
Allergan Inc.
2525 Dupont Dr.
Irvine, CA 92623
Phone (714)246-6572
E-mail: irenez@mail024.usirvine.allergan.sprint.com