%FREQ1VAR: Frequency of one variable with format: a macro to standardize proc FREQ output data sets

Ronald Fehd, Centers for Disease Control and Prevention, Atlanta GA

ABSTRACT

SAS(r) software provides the ability to associate formats, or look up tables, with variables in a data set. Encapsulating the format labels in a summary data set provides a complete and independent set of information about a variable to other procedures.

This macro was written to standardize summary data sets and bring them into conformance with data sets produced by Fehd's (1997) CHECKALL and SHOWCOMB macros, which provide information about multiple response data. The standardized data sets, or objects, have the same structure and are used for quick and concise reporting of summary information of large survey questionnaire data sets. Their identical structure enables easy access by other methods and software.

INTRODUCTION

Proc FREQ provides a summary of a variable; when a format is associated with the variable the procedure prints the value using its associated format. While a paper report is often sufficient, the increasing demand for graphics reports and presentations creates an expectation for an object that contains a complete set of information about a data set and the variable(s) being reported.

This macro processes a proc FREQ output data set and adds format labels and attributes to the data set. An attribute is a single item of information and includes the name of the data set against which the proc was run, the number of identifiers and number of observations of the data set, the number of valid responses in the variable and the response rate: the ratio of valid response to number of observations. These attributes are placed in variable labels and the first observation of the summary data set.

Setup: sets of values, autoexec, proc FORMAT

Observations in a data set contain one of three sets of values. When a variable is defined, its value is initialized to blank for character variables, or missing for numeric variables. As data entry or data manipulation occurs, variables receive valid values. In a small minority of cases, variables obtain invalid values.

The standard labels for blank/missing and invalid are set as global macro variables in the autoexec for the session. This allows the labels to be assigned with proc FORMAT in a separate program from the summarization program. The autoexec contains these statements:

```
%LET BLANK =BLANK;
%LET INVALID=INVALID;
```

During a data review process, it is necessary to report all three sets of data values, but for summary purposes, blank/missing and invalid values are expected to be excluded. This is accomplished in this macro by using standardized labels in proc FORMAT value statements. Valid values have their description. Character variables have space and dot labeled as "&BLANK.". Numeric variables have missing and in some cases, zero, labeled as "&BLANK.". Values to be excluded are grouped with the phrase: other="&INVALID.".

A program which contains formats would have these statements:

```
proc FORMAT;
value Num 1 ='one'  2 ='two'
 . ="&BLANK." other="&INVALID." ;
value $Chr '1'='one''2'='two'
'' ="&BLANK." other="&INVALID." ;
```

Refer to the test data at the end of the macro for further examples.

The process of standardization

This macro consists of several steps: initialization, using proc FREQ to produce an output data set, making attribute data sets, reading the data to prepare the other attributes and finally writing the summary data set to the library,

In the initialization step, all local macro variables are declared, the output data-set name is initialized, if not provided as a parameter, and options for macro debugging are turned on or off.

Proc FREQ is the first major step in the macro. Data are excluded with a where statement which compares the formatted value of the variable to the macro variables BLANK and INVALID. Parameters are provided for an additional where clause with which to subset the data and to sort the output data set by descending Count.

The %NOBS macro provides the number of observations of the data set. The %NOBS macro is based on the %OBSNVARS macro (*SAS Macro* Language Reference, *First Edition).* If the output data set contains no observations, then the macro stops processing and returns a completion code of zero.

The second step consists of preparing two data sets which contain the attributes Number-of-Ids and Number-of-Observations. These data sets are merged into the final data set.

In the third step, the format labels are added to the data set. The width of the three major variables, Label, Count and Percent, are calculated. The number of responses is accumulated from Count.

The label of the analysis variable is copied to a macro variable in the fourth step; other macro variables are created containing the various widths and the number of responses. When the data step is complete then the percentage of response is calculated.

During the final data step, the attributes are placed in the respective variable labels as the summary data set is written to the library.

Summary data set and attributes: definitions

The FREQ1VAR macro provides several attributes in addition to the three variables of the proc FREQ data set.

N-IDS: Number of Identifiers: When reporting summary information, either the number of observations or the number of identifiers is typically presented. Any data set either is unique on its identifiers or has multiple occurrences of its identifiers, therefore the value of N-IDS is less than or equal to the number of observations. This item is stored in the first observation of a character variable.

N_OBS: Number of Observations with valid values: data may be excluded from the summary; therefore the number of observations

with valid values is saved. This item is stored in two places: as the first observation of this character variable, and in the label of that variable. Three other items are stored in the label as well: the name and total number of observations of the data set, and the response rate (Response / Total). Here is an example of the label:
N=5 data:TESTDATA Obs:15 Resp:33%

LABEL: the variable's value, from its format: a character variable whose length is determined by the widest label. The label of this variable is the label of the analysis variable.

COUNT: number of observations with this value, a numeric integer.

PERCENT: of observations with this value, a numeric real; note that the denominator is the sum of Count.

VALUECHR, or VALUENUM: the analysis variable, renamed according to its type. This variable is provided in the data set as a check that the correct format was applied.

TITLE: to be used, for graphics or other presentation, if not the variable label, or where the length is greater than 40 characters.

_BY_VAR: names of variable(s) used in a cross-tabulation are stored in the label of this character variable. There are novalues in rows of the summary data set.

-SUBSET: additional where clause used to subset the data set before processing is stored in the label of this character variable. There are no values in rows of the summary data set.

CONCLUSION

Proc FREQ provides an output data set that contains three variables: the analysis variable, and Count and Percent. This macro renames the analysis variable, adds a character variable with the format label, calculates and adds attributes to the data set to create a package of information in a standardized object that can be used by other methods and applications for graphics and presentations.

REFERENCES

Fehd, Ronald (1997), "%CHECKALL, a macro to produce a frequency of response data set from multiple-response data," *Proceedings of the Twenty-Second Annual SAS Users Group International, 22: 1084.*

Fehd, Ronald (1997), "%SHOWCOMB: a macro to produce a data set with frequency of combinations of responses from multiple-response data," *Proceedings of the Twenty-Second Annual SAS Users Group International, 22: 939.*

SAS Institute Inc. (1997), *SAS" Macro Language Reference, First Edition,* Cary, NC: SAS Institute Inc.

Author: Ronald Fehd e-mail: RJF2@cdc.gov
Centers for Disease Control MS-G25
4770 Buford Hwy NE
Atlanta GA 303413724 voice: 770/488-8102
SAS-L archives: send e-mail
to: SAScontrib@SASserv.uga.edu
for %FREQ1VAR subject: cntb0038: download
for %CHECKALL subject: cntb0032: download
for %SHOWCOMB subject: cntb0033: download

ACKNOWLEDGMENTS

This routine was developed over a period of ten years while I crunched the numbers of survey data collected by the Model Performance Evaluation Program (MPEP) of the Division of Laboratory Systems, Public Health Practice Program Office of the Centers for Disease Control and Prevention, Atlanta, Georgia. John Hancock, chief, Information Services Activity, encouraged me to write up these routines. I wish to thank Sharon Blumer, David Cross, Thomas Hearn, and William Schalla of the MPEP group for their perseverance while I developed this routine.

```
 /*RJF2.SAS.MACROS(NOBS) ----------------------------------------
from SAS Guide to Macro Processing, V6, 2nd sd., pg 263
MACRO NOBS returns macro-var with user-supplied name, default==NOBS
which contains value of number of obs in most recently created SSD.
.....'
%MACRO NOBS(_MAC_VAR,DATA=.,_GLOBAL_=0)
/store dss = 'returns mac-var w/Nobs: SYSLAST or DATA' /*  ●  ,
;run;
%IF "&_MAC_VAR" = "" %THEN %LET _MAC_VAR = NOBS;
%IF &_GLOBAL_ %THEN %DO; %global &_MAC_VAR.;                    %END;
%IF &DATA = . %THEN %LET DSN = &SYSLAST.;%*resolve mac-vars in name;
%ELSE              %LET DSN = &DATA.;
%LET DSID = %sysfunc(open(&DSN));
%IF &DSID %THEN %DO;
 %LET &_MAC_VAR. = %sysfunc(attrn(&DSID,NOBS));
 %*LET NVARS     = %sysfunc(attrn(&DSID,NVARS));
 %*put NOBS returns nvars:"&NVARS"=<&&&NVARS>;
 %LET RC = %sysfunc(close(&DSID.));                             %END;
%ELSE %put Open for data set &DATA. failed - %sysfunc(sysmsg());
%put NOBS: "&_MAC_VAR"=<&&&_MAC_VAR> data=&DSN.;/*........*NOBS*/%MEND;

/* MACRO: FREQ1VAR returns summary data set            uses macro: NOBS
•                 from data set, variable and format
*  AUTOEXEC:  %LET BLANK =BLANK;
*            %LET  INVALID-INVALID;
*            %LET DATA_SET=<data-set-name>;
• USAGE:  1) %FREQ1VAR(var-name,format);
*         2) %FREQ1VAR(var-name,format,OUT=ABC);
•         ■> %FREQ1VAR(var-name,format,PRINT=1);
*         4) %FREQ1VAR(var-name,format,TESTING=1);
*         5) %FREQ1VAR(var-name,format,GRFXPTRN=PIE);
*         6) %FREQ1VAR(var-name,format,ORDER=DATA);
*         7) %FREQ1VAR(var-name,format,WHERE=Var1 eq 'A');
*            do not "se comparison operator   symbols: =
*                    use comparison operator mnemonics: eq
*  PROCESS:
*  0. setup and initialization
*  1. proc FREQ
·     where: exclusion of BLANK, INVALID
•             other subset -- where clause -- if present
*     %NOBS: if output data set is empty then exit
*     sort, if wanted
*  2. make attributes: N_IDS and N_OBS
*  3. read summary dsta
•     make Label "sing format
*     if Colon in Label front-trim Label
•     if Graphics-Pattern = PIE append Count+Percent to Label
•     calculate max lengths, accumulate Count to N_Resp
*  4. make mac-vars: VarLabel N_Resp
•           nax-length of: Label, Count, Psrcsnt
      calculate %-Resp
*  5. make output data sst
*     rename Variable: ValueChr or ValueNum
*     place attributes in variable labels
*  6. if PRINT or TESTTNQ: Print output data set + contents
• NOTES:
• • ×□□▱• name must contain dot as suffix
* * mac-vars BLANK and INVALID usually defined in autoexec
*     so format Program and macro can access them as global variables
• • variable labels nay have colons, if so, front-trim to colon
*     e.g.: 'Q06: Supervisor'; change to: 'Supervisor'
* KEYWORDS: autoexec formats %NOBS FAEQ object standardization
* CONTENTS of output data set:
• ▱ N_IDS     Char  4  $CHAR4.  N Ids
* 2 N_OBS     Char  4  $CHAR4.  N=5 data:TESTDATA Obs:15 Resp:33%
* 3 LABEL     Char  5  $CHAR5.  Num three levels
* 4 COUNT     Num   4  1.       # of IDs Responding
* 5 PERCENT   Num    8 4.1     % of IDs Responding
* either of:
• 6 VALUENUM  Num   4  N1_3_.   N1_3 value
* 6  VALUECNR  Char  1  C1_3_.   C1_3 value
• 7 TITLE     Char 16  $CHAR16. Title
* 8 _BY_VAR   Char  1  $CHAR1. by_var: .
• 9 _SUBSET  ▫▱▫  1 $CHAR1. subset: ▱
* AUTHOR: Ronald Fohd                              s-mail: RJF2@cdc.gov
*         Centers for Disease Control, and Prevention
*         4770 Buford Hwy NE MS-025                  fax: 770/488-8282
•         Atlanta GA 30341-3724                    voice: 770/488-8102*/
%MACRO FREQ1VAR(/* * * * * * * * * * * * * * * * * * * * * * * * * * */
```

```
 VARIABLE /* variable name                                              */
,FORMAT   /* format of variable with suffix=dot                        */
,DATA     =&DATA_SET./* DATA_SET is global variable, else hardcode here*/
,ID       =IdNmbr   /* var for N_IDS                                   */
,_BLANK   =&BLANK.  /*common format label indicating blank/missing     */
,_INVALID=&INVALID./*common format label indicating out of range       */
,GRFXPTRN=BARH/* graphics-pattern in (barh barv pie)                    */
,LIBRARY =WORK    /* library name for read and write ,LIBRARY =LIBRARY **
,LIBRARY =LIBRARY/* library name for read and write ,LIBRARY =WORK    */
,LBL_VAR =./* label of variable, see %local VARLABEL                  **
           /* provide to overwrite &DATA_SET label of &VARIABLE       */
,LBL_CNT =Number of Laboratories Responding/*                          **
           /* label of frequency count                                */
,LBL_PCT =Percentage of Laboratories Responding/*                      **
           /* label of frequency percent                              */
,MISSING =0/* ?include BLANK/MISSING in FREQ?                         */
,OUT      =./* output data set name, if not &VARIABLE.                */
,ORDER    =FREQ/* ?sort descending Count?, else ORDER=DATA            */
,PRINT    =0/* ?print output data_set? ,PRINT=1                       */
,BY_VAR   =./* by_var for subsetting                                  */
,WHERE    =./* where statement for subsetting                         **
           /* where=var1 eq 'A'                                       **
           /* where=var1 eq 'B' and var2 eq 'A'                       **
           /* where=var1 eq 'B' and not var2 eq 'A'                   */
,TITLE    =./* title for graphics default: &VARIABLE label           */
,TESTING =1/* ?print intermediate data-sets and messages?  ,TESTING=0 */
)/store des='FREQ1VAR: freq of one var w/format'/**********************/
;%*0: setup;
%global FREQ1VAR;%LET FREQ1VAR = 1;%*summary data set created;
%local LEN   LENCOUNT LENLABEL LENN_OBS LENPCENT
       N_IDS N_OBS    N_RESP   PCNTRESP VARLABEL;
%IF "&OUT." eq "." %THEN %LET OUT = &VARIABLE.;
%IF &TESTING %THEN %DO; %LET PRINT=1;
                        options   mprint   notes;                    %END;
 %ELSE                %DO; options nomprint nonotes;                 %END;

%*1;proc FREQ data = &LIBRARY..&DATA.
 (where=(put(&VARIABLE.,&FORMAT.) not in("&_BLANK","&_INVALID")
  %IF "&WHERE"  ne "." %THEN and &WHERE. ;          ) ) %*proc closure; ;
 format &VARIABLE. &FORMAT.;
 tables
  %IF "&BY_VAR." ne "." %THEN &BY_VAR. *;
  &VARIABLE. / out = FREQ1VAR noprint
  %IF &MISSING %THEN missing;                          %*tables closure; ;

%IF &TESTING %THEN %DO;proc PRINT data = FREQ1VAR;                   %END;
%NOBS(N_OBS);run;
%IF not &N_OBS %THEN %DO;%LET FREQ1VAR = 0;%*summary data NOT created;
                        %PUT @@@FREQ1VAR:obs=0 for &VARIABLE. &FORMAT.;
                        %GOTO ENDOMAC;                               %END;

%IF &ORDER=FREQ %THEN %DO;
 proc SORT data = FREQ1VAR;
  by %IF "&BY_VAR." ne "." %THEN &BY_VAR.;
     descending Count;                                               %END;

%*2: make attributes N_OBS and N_IDS;
%NOBS(N_OBS,DATA=&LIBRARY..&DATA.);run; %LET LENN_OBS = %length(&N_OBS);

%*make N_OBS, see also in FREQXTAB SHOWCOMB;
proc MEANS data = FREQ1VAR noprint;
 var Count;
 output sum=Count out=N_OBS(drop=_Type_ _Freq_);
 %IF  "&BY_VAR." ne "." %THEN %DO;                                   %END;
  by   &BY_VAR.;

DATA    N_OBS(drop = Count);
 length N_Obs $ %eval(&LENN_OBS. +2);
 do until(EndoFile);
  set   N_OBS end = EndoFile;
        N_Obs = 'N=' || trim(left(put(Count,&LENN_OBS..)));
  output; end; stop;

%*make N_IDS;
proc FREQ data = &LIBRARY..&DATA.
 (where=(put(&VARIABLE.,&FORMAT.) not in("&_BLANK","&_INVALID")
  %IF "&WHERE"  ne "." %THEN and &WHERE. ;          ) ) %*proc closure; ;
 tables
  %IF "&BY_VAR" ne "." %THEN &BY_VAR. *;
  &ID. / noprint out = FREQ_ID;

%IF "&BY_VAR" eq "." %THEN %DO;
 %NOBS(N_IDS);run;DATA N_IDS;Count = &N_IDS;output;stop;             %END;
%ELSE %DO;
 proc FREQ data = FREQ_ID;
  tables &BY_VAR / list noprint out = N_IDS(drop=Percent);           %END;

DATA    N_IDS;
 length N_Ids $ %eval(2 + &LENN_OBS.);
 do until(EndoFile);
  set   N_IDS end = EndoFile;
        N_Ids = 'N=' || trim(left(put(Count,&LENN_OBS..)));
  output; end; stop;
```

```
 attrib C1_4 length=$1 format=$C1_4_. label='Chr four directions'
        CA_C length=$1 format=$CA_C_. label='Chr fruit list'
        N1_3 length= 4 format= N1_3_. label='Num three levels'
        N1_9 length= 4 format= N1_9_. label='Num three ranges'
 ;do N1_9 = 0 to 9; N1_3 = N1_9; C1_4 = put(N1_9,1.);
   IdNmbr=put(N1_9,z2.); byVar = (N1_9 le 5);
   if N1_9 then CA_C = substr('ABCDEFGHI',N1_9,1);
   if mod(N1_9,2) then output; output; end; stop;
 proc PRINT data = TESTDATA;format _all_;proc CONTENTS data = TESTDATA;
  /*********************************************************************
```

```
%*3;DATA FREQ1VAR;
 length Label VarLabel $ 40;
 retain LenLabel LenCount LenPcent N_Resp 0;
 drop   LenLabel LenCount LenPcent N_Resp VarLabel Colon;
 do until(EndoFile);%*-----------------------------------------------*;
  set FREQ1VAR  end = EndoFile;
  Label = left(put(&VARIABLE,&FORMAT.));
  %*if colon present, front-trim to colon;
  Colon = index(Label,':');
  if Colon then Label = left(substr(Label,Colon+1));
  %IF %index(%upcase(&GRFXPTRN.),PIE) %THEN %DO;
   %*append "<new-line> Count (Percent%)" to Label;
  Label   = trim(Label) || '^'  || trim(left(put(Count,&LENN_OBS..0)))
                        || ' (' || trim(left(put(Percent,5.1)))
                        || '%)';                                 %END;
  LenLabel=max(LenLabel,length(              Label                 ));
  LenCount=max(LenCount,length(trim(left(put(Count  ,&LENN_OBS..0)))));
  LenPcent=max(LenPcent,length(trim(left(put(Percent,5.1          )))));
  N_Resp + Count;
  output; %*.................................. *do until(EndoFile); end;
%*4 make mac-vars;
 call label(&VARIABLE.,VarLabel);
 %*change squote to explanation mark to avoid mac-error;
 VarLabel = translate(VarLabel,'!',"'");
 %*if colon present, front-trim to colon;
 Colon = index(VarLabel,':');
 if Colon then VarLabel = left(substr(VarLabel,Colon+1));
 call symput('VARLABEL',trim(left(    VarLabel              )));
 call symput('N_RESP'  ,trim(left(put(N_Resp  ,&LENN_OBS..))));
 call symput('LENLABEL',trim(left(put(LenLabel,2.          ))));
 call symput('LENCOUNT',trim(left(put(LenCount,&LENN_OBS..))));
 LenPcent = max(LenPcent,3);%*kludge for missing with Percent=.;
 call symput('LENPCENT',trim(left(put(LenPcent,5.          ))));stop;run;

%LET PCNTRESP = %eval(100* &N_RESP /&N_OBS);
%IF "&LBL_VAR" eq "." %THEN %LET LBL_VAR = &VARLABEL.;
%IF "&TITLE"   eq "." %THEN %LET TITLE   = &VARLABEL.;
%IF &TESTING %THEN %DO;proc PRINT data = FREQ1VAR;
                  %put PCNTRESP=<&PCNTRESP.>LENLABEL=<&LENLABEL.>;%END;
%*5: make output data set;
DATA &LIBRARY..&OUT.(label = "FREQ1VAR: &VARIABLE. fmt: &FORMAT."
                    rename=(&VARIABLE =
 %IF "%substr(&FORMAT.,1,1)" = "$" %THEN ValueChr;
 %ELSE                                   ValueNum;  %*DATA closure:";));
 attrib    %LET           LEN  = %eval(2 + &LENN_OBS.);
  N_Ids    length = $    &LEN.
           format = $char&LEN..      label = "N Ids"
  N_Obs    length = $    &LEN.
           format = $char&LEN..      label =
                    "N=&N_RESP data:&DATA Obs:&N_OBS Resp:&PCNTRESP.%"
  Label    length = $    &LENLABEL.
           format = $char&LENLABEL.. label = "&LBL_VAR."
  Count    length = 4
           format = &LENCOUNT..0     label = "&LBL_CNT."
  Percent  format = &LENPCENT..1     label = "&LBL_PCT."
  &VARIABLE                          label = "&VARIABLE. value"
           %LET           LEN =%length(&TITLE.);
  Title    length = $    &LEN.      label = "Title"
           format = $char&LEN..
  _By_var  length = $    1
           format = $char1.          label = "by_var: &BY_VAR."
  _Subset  length = $    1
           format = $char1.          label = "subset: &WHERE.";
 retain Title "&TITLE." _By_var _Subset ' ';
 do until(EndoFile);
  merge N_IDS N_OBS FREQ1VAR  end = EndoFile;
  %IF "&BY_VAR." ne "." %THEN %DO;
  by &BY_VAR.;                                                       %END;
  output; Title='.'; %*.................. do until(EndoFile)*; end; stop;

%*6;%IF &PRINT or &TESTING %THEN %DO;
 proc PRINT data = &LIBRARY..&OUT. double label noobs;
  format _all_;                                                      %END;
%IF &TESTING %THEN %DO;
 proc CONTENTS data = &LIBRARY..&OUT.;                               %END;
%ENDOMAC:run; %*.....................................................*; %MEND;
 /*test data*************************************************************
%*autoexec;%LET DATA_SET=TESTDATA;%LET BLANK=BLANK;%LET INVALID=INVALID;
*libname LIBRARY '<lib-ref>';
proc FORMAT;
 value $C1_4_ '1'='C1-4:one-North  '
              '2'='C1-4:two-East'
              '3'='C1-4:three-South'
              '4'='C1-4:four-West' ' ','.'="&BLANK." other="&INVALID.";
 value $CA_C_ 'A'='CA-C:apple '
              'B'='CA-C:banana'
              'C'='CA-C:cherry'    ' ','.'="&BLANK." other="&INVALID.";
 value N1_3_  1 = 'N1-3: one  '
              2 = 'N1-3: two'
              3 = 'N1-3: three'         .,0="&BLANK." other="&INVALID.";
 value N1_9_  1 -  4  = 'N1-9: 1..4
              5 -  9  = 'N1-9: 5..9'
              9<- high= 'N1-9: >9'      .,0="&BLANK." other="&INVALID.";
data TESTDATA;
```

```
 %FREQ1VAR(N1_3, N1_3_.,TESTING=1);
 %FREQ1VAR(N1_9, N1_9_.,BY_VAR=ByVar);
 %FREQ1VAR(C1_4,$C1_4_.);
 %FREQ1VAR(CA_C,$CA_C_.);
 %FREQ1VAR(N1_3,N1_3_.,GRFXPTRN=PIE);
 %FREQ1VAR(N1_3,N1_3_.,ORDER=DATA);
 %FREQ1VAR(N1_3,N1_3_.,WHERE=N1_3 gt 3);%*obs=0 msg;
  /* END TEST SECTION ..................................................*/
```