Paper 033-30

Using Variable Values to Change Macro Variables in Loop Processing

Michael G. Eberhart, Philadelphia Department of Public Health, Philadelphia, Pennsylvania

Keywords: ODS, macro variables, CALL routine

Abstract

SAS® and the Output Delivery System (ODS) can make printed or web-based output processing easier and more productive. Creating multiple reports from a single dataset can be repetitive and time-consuming. Using macro variables and variable formats can simplify this process. Using macro processing and one data step, macro variable values can be used to name output files, limit observations in the dataset, and add titles to individual reports. The CALL routine can be used to facilitate this process.

Introduction

When creating multiple reports from a single dataset, it is often necessary to create individually named and titled reports. SAS® and the Output Delivery System (ODS) make the process much easier and the reports more consistent. Whether creating .rtf or .pdf documents, or HTML files for the web, these reports are often similar in regard to table layout and style, with modest differences in the report title. As an example, cancer incidence data is often separated by cancer site (e.g., lung, breast, colon) and patient demographic (e.g., white females, black males, etc.), and reports are often requested for specific cancer sites and demographics. The values of the cancer site and demographic variables (site, sex and race) can be used in macro processing to create individually named and titled reports based on the variables in the analysis.

Data

For the purposes of this paper, assume that a dataset has been created (work.cntyrates) that includes the county of residence (resco), number of cases (count), age-adjusted rate (aarate), lower confidence interval (lci) and upper confidence interval (uci) for all cancer sites broken down by gender and race. The confidence interval variables will be recoded into character variables (lcichr and ucichr) so that they could be combined into one character variable (confint) that describes the range of the confidence interval. There are 45 major cancer site groups (plus an additional group that includes all cancers) and twelve race/sex categories, for a total of 552 different subgroup combinations (46*12=552). The required output is an HTML file for each subgroup for presentation on the organization’s website.

Output Table

The table is defined in the following PROC TEMPLATE procedure. The columns and headers are defined in the DEFINE TABLE statements. Pre-formatted text is used in the DEFINE HEADER statements to allow for greater flexibility in column width. The confidence interval variables were included in the template, and the PRINT=OFF statements prevents them from displaying in the output. Only the character variable, confint, is used in the report.

```
proc template;
define table ratesbycounty;
     print_footers=off;
     print_headers=off;
     column resco count aarate lcichr ucichr confint;
     define resco;
          header=h4;
          just=left;
          vjust=center;
          format=coa.;
          format_width=32;
```

```
            define header h4;

            text'   Rate';
            preformatted=on;
            style=headingFont{font_size=9pt};
            end;
end;
define lcichr;
      print=off;
      drop=on;
end;
define ucichr;
      print=off;
      drop=on;
end;
define confint;
      header=h3;
      format=$25.;
      vjust=center;
      just=c;
      width=25;
      define header h3;
            text' Confidence Interval';
            preformatted=on;
            style=headingFont{font_size=9pt};
            end;
end;
end;
run;
```

The template procedure was also used to define a new style for this table presentation, using the *styles.printer* template as a parent. Aside from the fonts and cellpadding, the style element with the most impact on the report layout is the RULES=GROUPS and FRAME=ABOVE statements. These elements determine where ODS draws lines in the table and how the output is framed.

```
proc template;
   define style cancer.rptstyle2;
      parent = styles.Printer;
      replace fonts /
         'TitleFont2' = ("Times Roman",9pt,Bold)
         'TitleFont' = ("Times Roman",10pt,Bold)
            'FootnoteFont'=("Times Roman",8pt)
         'StrongFont' = ("ITC Bookman, Times Roman",10pt,Bold)
            'EmphasisFont' = ("ITC Bookman, Times
      Roman",10pt,Italic)
         'FixedEmphasisFont' = ("Courier",9pt,Italic)
         'FixedStrongFont' = ("Courier",9pt,Bold)
         'FixedHeadingFont' = ("Courier",9pt,Bold)
         'BatchFixedFont' = ("SAS Monospace, Courier",7pt)
         'FixedFont' = ("Courier",8pt)
         'headingEmphasisFont' = ("Times Roman",8pt)
         'headingFont' = ("Times Roman",8pt)
         'docFont' = ("Times Roman",7.5pt);
      style Table from Output /
             background = _undef_
             rules=groups
         frame = ABOVE
      rightmargin = _undef_
      leftmargin = _undef_;
      replace HeadersAndFooters from Cell /
      borderwidth=.75pt;

 replace Body from Document
         "Controls the Body file." /
         bottommargin =.25in
         topmargin = .25in
         rightmargin = .25in
         leftmargin = .25in;
       style SysTitleAndFooterContainer from Container
         "Controls container for system page title and system
page footer." /
         rules = NONE
         frame = VOID
         outputwidth = _undef_
         cellpadding = 0
         cellspacing = 0
         borderwidth = 0;
      style TitleAndNoteContainer from Container
         "Controls container for procedure defined titles and
notes." /
         rules = NONE
         frame = VOID
         outputwidth = _undef_
         cellpadding = 0
         cellspacing = 0
         borderwidth = 0;
      style TitlesAndFooters from Container
         "Abstract. Controls system page title text and system
page footer text." /
         font = Fonts('TitleFont2')
         background = colors('systitlebg')
         foreground = colors('systitlefg');

      style SystemTitle from TitlesAndFooters
         "Controls system title text." /
         font = Fonts('TitleFont');
      style SystemFooter from TitlesAndFooters
         "Controls system footer text." /
         font = Fonts('TitleFont');
      style PageNo from TitlesAndFooters
         "Controls page numbers for printer" /
         cellspacing = 0
         cellpadding = 0
         font = fonts('strongFont')
             foreground=white;
   end;
run;
```

Formats for variables used in the table were also created. The values for these formats will be used to create additional variables and supply values for macro variables in the report processing.

```
proc format;
VALUE sitenewb 0="All Cancer" 1="Lip" 2="Tongue" 3="Major
Salivary Gland" 4="Floor of Mouth" 5="Gum and other Mouth"
6="Nasopharynx" 7="Oropharynx" 8="Hypopharynx"
9="Esophagus" 10="Stomach" 11="Small Intestine" 12="Colon
excl. rectum" 13="Rectum and rectosigmoid"
14="Anus" 15="Liver" 16="Gallbladder" 17="Pancreas"
18="Larynx" 19="Lung and Bronchus" 20="Bones and Joints"
21="Soft Tissue" 22="Melanomas" 23="Breast" 24="Cervix"
25="Corpus Uteri" 26="Uterus NOS" 27="Ovary" 28="Prostate"
29="Testis" 30="Penis" 31="Bladder" 32="Kidney"
33="Ureter" 34="Eye" 35="Brain" 36="Thyroid Gland"
37="Hodgkins Disease" 38="Non-Hodgkins Lymphoma"
39="Multiple Myeloma" 40="Acute Lymphocytic Leukemia"
41="Chronic Lymphocytic Leukemia" 42="Acute Myeloid
Leukemia" 43="Chronic Myeloid Leukemia" 44="Other
Leukemias" 45="Other Sites";
run;

proc format;
value rsa 0='All Races and Genders' 1='Males' 2='Females'
10='Whites' 11='White Males' 12='White Females' 20='Blacks'
21='Black Males' 22='Black Females'30='Other Race'
31='Other Males' 32='Other Females';
run;
```

Data Preparation

In order to limit the number of macro variables required, the race and sex variables are combined into a single variable (rs). In this example, the values for sex are 1 for male and 2 for female. The values for race are 1,2 and 3 representing white, black and other, respectively. For white males the calculation would be (10*1)+1, or 11. This new variable (rs) will be used to create another variable called 'category' that will be used to limit the dataset during macro processing.

```
data cntyrates1; set cntyrates;
rs=(10*race)+sex;
run;
```

Using the variable formats defined above, the PUT function is used to create text variables from numeric variables, allowing the values to be used as macro variables. Counts of less than 6 are suppressed for privacy reasons, and the rates are set to missing. Also, the variables lcichr (lower confidence interval) and ucichr (upper confidence interval) are combined for presentation into a new text variable (confint) using the concatenation operator. The variables sitename and category will be used in a data step to control the macro processing.

```
data cntyrates2; set cntyrates1;
sitename=put(site,sitenewb.);
category=put(rs,rsa.);
if count lt 6 then aarate=.;      /*Suppress rates if count lt 6*/
if count gt 5 then confint=put('('|| lcichr ||'  to '|| ucichr
||')',$25.);
if 1 le count le 5 then confint=' ';
/*Set missing count to zero*/
if count=. then count=0 and confint=' ';
run;
```

The LET statement creates a macro variable to set the output pathname.

```
%LET OUTPUT1 =C:\Cancer\OUTPUT\webpage\1997to2000\;
```

The macro “Reports” includes two macro variables – sitename and category – that will be used to name the output file and complete the titles of each individual report. The ODS escapechar is used to allow for special characters, in this case superscript for footnotes, to be included in the titles and footnotes. The data step in the macro uses the macro variables to limit the observations included in the FILE PRINT statement.

```
%macro reports(sitename,category);
ods listing close;
options nodate orientation=landscape;
ods html file="&OUTPUT1.&sitename._&category..html"
style=cancer.rptstyle2;
ods escapechar='\';
title1 'Total Cancer Cases\{super 1} and Age-Adjusted Cancer
Incidence Rates\{super 2}';
title2 "&sitename by County, Tennessee, 1997-2000, &category ";
footnote1 font="Times Roman" height=8pt j=l 'Source:  Tennessee
Cancer Registry.  Data are approximately 80% complete. Interpret
with caution.';
footnote2 font="Times Roman" height=8pt j=l '\{super 2}Counts and
rates are suppressed when fewer than 6 cases were reported';
footnote3 font="Times Roman" height=8pt j=l '\{super 1}Rates are
per 100,000 and are age-adjusted using the 2000 U.S. population
standard';

data _null_; set cntyrates2;
    if sitename="&sitename";
    if category="&category";
file print ods=(template='ratesbycounty');
drop sitename;
put _ods_;
run;

ods html close;
%mend;
```

Prior to execution of the data step, the dataset is sorted on the BY variables used in processing.

```
proc sort data=cntyrates2;by sitename category;run;
```

Execution of the data step utilizes a CALL routine to feed the values of the two sorted variables to the execution of the macro. The data step is executed only once, but the call routine will execute 552 times; once for each unique combination of sitename and category.

```
data _null_; set cntyrates2;
  by sitename category;

  if first.sitename then do;
  tot+1;
  end;
      if first.category then do;
      cat+1;
      call execute('%reports('||sitename||','||category||')');
      end;
run;
```

The execution of the data step creates 552 individual HTML files that are all stored in a common folder. The file names and titles of the reports are all consistent and the tables are identical. This routine simplifies the process of creating multiple reports from a single dataset, and in conjunction with ODS processing allows for flexibility in naming the output files and title statements for the tables.

REFERENCES

SAS Institute Inc., *The Complete Guide to the SAS® Output Delivery System, Version 8,* Cary, NC: SAS Institute Inc., 1999. 310 pp.

CONTACT INFORMATION

Comments and questions are valued and encouraged. Contact the author at:

Michael G. Eberhart, MPH
Philadelphia Department of Public Health
500 S. Broad St.
Philadelphia, PA 19146
215-685-6603
Michael.eberhart@phila.gov