

Automated Comparison Tables using PROC REPORT

Evan Kwiatkowski, University of North Carolina at Chapel Hill

ABSTRACT

Comparison tables showing rows of summary statistics for continuous and categorical variables (e.g. age and gender) across columns of a categorical variable (e.g. treatment vs. placebo) are ubiquitous in clinical research. Many SAS® procedures are used to generate these data summaries and associated tests of statistical significance (e.g. PROC FREQ, PROC TTEST). Other pertinent information, such as formatted variable names and number of non-missing observations per variable, are often included in the final report. This paper presents a one-step macro for producing reports that contain summary statistics, tests for significance, and other relevant information in a comparison table in RTF format.

INTRODUCTION

A report with descriptive statistics is a foundational step in any statistical analysis. These reports involve two time-consuming components: (1) assembling the output from an array of SAS procedures and (2) formatting the results in a way that is presentable for dissemination. This macro automates both these components, providing a one-step solution to a time-consuming undertaking.

TABLE1 MACRO

The following arguments are used in the %table1 macro:

Macro Argument	Description
Dataset	Name of dataset to be used in the analysis.
Directory	Directory of dataset to be used in the analysis.
Response	The categorical response variable to appear in separate columns.
Response_format	The format of the categorical response variable.
Predictors_all	The full list of predictor variables in the order which they are desired to appear in the rows of the final report.
Predictors_countable	The list of predictor variables which will be presented with median, first quartile, and third quartile summary statistics and a p-value based on the Kruskal-Wallis test for median.
Predictors_continuous	The list of predictor variables which will be presented with mean and standard deviation summary statistics and a p-value based on ANOVA.
Predictors_categorical	The list of predictor variables which will be presented with n (%) output summary statistics and a p-value based on a chi-square test of homogeneity.
Categorical formats	The formats of the categorical predictor variables.

Table 1. Description of macro arguments

This macro is specified with a single categorical response variable with two or more categories. Varying numbers of predictors of each type are acceptable as arguments separated by a space. Each predictor variable must appear exactly twice: once in the Predictors_all argument in the order which it is to appear in the report and once in the appropriate predictor type argument (Predictors_countable, Predictors_continuous, or Predictors_categorical).

This macro was created with the response and predictors having numeric values in the dataset, and with each categorical variable having a corresponding user-specified format. It is the value of the format that is used in the formatted report.

EXAMPLE

This macro is demonstrated with the BWeight data set in the SasHELP directory, which provides data from the National Center for Health Statistics.

The following formats are used in the report:

```
proc format;
value education
  0 = 'High School'
  1 = 'Some College'
  2 = 'College'
  3 = 'Less Than High School';
value noyes
  0 = 'No'
  1 = 'Yes';
value pvalue_best
  0-<0.001='<0.001'
  0.001-<0.005=[5.3]
  0.005-<0.045= [5.3]
  0.045-<0.055=[5.3]
  other=[5.3];
value percent_best
  0-<0.5='<1'
  other=[3.0];
run;
```

The execution of the macro is as follows:

```
%table1 (
dataset=bweight,
directory=sasHELP,
response=MomEdLevel,
response_format=education,
predictors_all=weight momsmoke cigsperday,
predictors_countable=weight,
predictors_continuous=cigsperday,
predictors_categorical=momsmoke,
categorical_formats=noyes);
```

	Overall (N=50,000)	High School (N=17,449)	Some College (N=12,129)	College (N=12,449)	Less Than High School (N=7,973)	p-value
Infant Birth Weight	3402 (3062,3720)	3357 (3033,3700)	3430 (3090,3742)	3487 (3175,3799)	3289 (2948,3620)	<0.001
Smoking Mother						
No	43,467 (87%)	14,484 (83%)	10,982 (91%)	12,137 (97%)	5,864 (74%)	<0.001
Yes	6,533 (13%)	2,965 (17%)	1,147 (9%)	312 (3%)	2,109 (26%)	
Cigarettes Per Day	1.48 ± 4.65	1.90 ± 5.14	1.01 ± 3.79	0.25 ± 1.92	3.18 ± 6.66	<0.001

Output 1. Output from %table1 macro execution

The column headers and their sample sizes are automatically generated based on their frequencies in their datasets and the specified response variable format. An “Overall” column with the summary statistics for each predictor variable in the entire sample appears first. For categorical predictors, the row ordering specifies values with the lowest unformatted numeric value to appear first, and it is this row which

displays the p-value. The labels for the predictor variables as appear in the rows are taken directly from the metadata for the specified analysis dataset.

CONCLUSION

This macro provides a one-step automated process for a dissemination-ready dataset that includes summary statistics for predictor variables across levels of a categorical response variable, and a p-value based on tests of univariate association between each predictor variable separately and the response variable.

While much of the output is already formatted based on the variable labels and formats, the ordering of the categorical response level as appearing in the columns and the ordering of each categorical predictor variable as appearing in the rows is pre-specified to be in the order of the unformatted numeric variables. Based on Output 1, the user might wish to place the “Less Than High School” column between “Overall” and “High School” – this change would have to be made manually to the formatted table after macro execution.

REFERENCES

Kukhareva, Polina; Roggenkamp, Kathy. “Customize Comparison Tables for Clinical Studies.” SAS Conference Proceedings: Western Users of SAS Software 2013, November 13-15, 2013, Las Vegas, Nevada.

ACKNOWLEDGMENTS

This paper was based on macros developed at the Collaborative Studies Coordinating Center at the University of North Carolina at Chapel Hill.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Evan Kwiatkowski
ekwiatkowski@unc.edu

MACRO CODE

```
%macro
    table1(dataset,directory,response,response_format,predictors_all,predictors_cou
    ntable,
    predictors_continuous,predictors_categorical,categorical_formats);

data _null_; call symput('B',trim(left(input("A0",$hex2.)))); run; %let c=&b&b&b;

/*** LOOP OVER ALL PREDICTORS ***/
%do i=1 %to %sysfunc(countw(&predictors_all));
%let predictor=%scan(&predictors_all,&i);

/*** CATEGORICAL PREDICTORS ***/
%if %eval(%sysfunc(findw(&predictors_categorical,&predictor," ",E))>0) %then %do;
%let
format=%scan(&categorical_formats,%sysfunc(findw(&predictors_categorical,&predictor,"
",E)));
/* summary statistics */
ods output CrossTabFreqs=ProcFreq&predictor;
proc freq data=&directory.&dataset; tables &predictor*&response; run;
ods output close;
data ProcFreq&predictor; set ProcFreq&predictor;
if _TYPE_ =10 then &response=-99; /* -99 for Overall */
select;
```

```

when (_TYPE_=11) Freqdisplay=cat(put(FREQUENCY,comma8.), "
(",strip(put(COLPERCENT,percent_best.)), "%)");
when (_TYPE_=10) Freqdisplay=cat(put(FREQUENCY,comma8.), "
(",strip(put(PERCENT,percent_best.)), "%)");
otherwise
    Freqdisplay="";
end;
run;
proc sort data=ProcFreq&predictor; by &predictor &response; run;
proc transpose data=ProcFreq&predictor out=Freqs&predictor prefix=frequency;
by &predictor; id &response; var Freqdisplay; where &response NE .; run;
proc sql;
select label into :label from sashelp.vcolumn
where libname="%upcase(&directory)" and memname="%upcase(&dataset)" and
upcase(name)="%upcase(&predictor)";
quit;
data Freqs&predictor; length predictor $98; drop _Name_;
set Freqs&predictor(rename=(&predictor=level));
predictor = upcase("&predictor");
select;
when (level=.)          characteristic="&label";
otherwise
    characteristic=cat("&c",put(level,%sysfunc(cat(&format,.)))));
end;
run;
proc sql;
select min(level) into :min_level from freqs&predictor;
quit;
/* significance test */
proc freq data=&directory.&dataset;
tables &response*&predictor/chisq;
output out=p&predictor pchi;
run;
data p&predictor(keep=predictor P_PCHI level); length predictor $98; set p&predictor;
predictor=upcase("&predictor");
level=&min_level;
run;
/* merge summary and significance test */
data final&predictor;
merge Freqs&predictor p&predictor; by predictor level;
run;
proc append data=final&predictor base=final force; run;
proc datasets nolist;
delete ProcFreq&predictor Freqs&predictor p&predictor final&predictor;
quit;
%end;

/** CONTINUOUS PREDICTORS **/
%if %eval(%sysfunc(findw(&predictors_continuous,&predictor," ",E))>0) %then %do;
/* summary statistics */
proc sort data=&directory.&dataset; by &response; run;
proc means data=&directory.&dataset; by &response; var &predictor;
ods output summary=summarya&predictor;
run;
proc means data=&directory.&dataset; var &predictor;
ods output summary=summaryb&predictor;
run;
data summaryb&predictor; set summaryb&predictor;
&response=-99;
run;
data summary&predictor; set summarya&predictor summaryb&predictor;
Freqdisplay=cat(put(&predictor._Mean,comma6.2), " %sysfunc(byte(177))
",strip(put(&predictor._StdDev,comma6.2)));
run;

```

```

proc transpose data=summary&predictor out=Freqs&predictor prefix=frequency;
id &response; var Freqdisplay;
run;
proc sql;
select label into :label from sashelp.vcolumn
where libname="%upcase(&directory)" and memname="%upcase(&dataset)" and
upcase(name)="%upcase(&predictor)";
quit;
data Freqs&predictor; length predictor $98; drop _Name_; set Freqs&predictor;
predictor = upcase("&predictor");
characteristic="&label";
level=.; /* avoids an error from proc append */
run;
/* significance test */
ods output overallanova=p&predictor;
proc anova data=&directory..&dataset;
class &response; model &predictor=&response;
run; quit;
ods output close;
data p&predictor(keep=predictor P_PCHI); set p&predictor(rename=(probf=P_PCHI));
predictor=upcase("&predictor"); where source='Model';
run;
/* merge summary and significance test */
data final&predictor;
merge Freqs&predictor p&predictor; by predictor;
run;
proc append data=final&predictor base=final force; run;
proc datasets nolist;
delete summarya&predictor summaryb&predictor summary&predictor freqs&predictor
p&predictor final&predictor;
quit;
%end;

/**** COUNTABLE PREDICTORS ****/
%if %eval(%sysfunc(findw(&predictors_countable,&predictor," ",E))>0) %then %do;
/* summary statistics */
proc sort data=&directory..&dataset; by &response; run;
proc means data=&directory..&dataset q1 median q3;
by &response; var &predictor;
ods output summary=summarya&predictor;
run;
proc means data=&directory..&dataset q1 median q3;
var &predictor;
ods output summary=summaryb&predictor;
run;
data summaryb&predictor; set summaryb&predictor;
&response=-99;
run;
data summary&predictor; set summarya&predictor summaryb&predictor;
Freqdisplay=catt(trim(put(&predictor._Median,comma4.)), "
(",trim(put(&predictor._Q1,comma4.)),",",trim(put(&predictor._Q3,comma4.)),")");
run;
proc transpose data=summary&predictor out=Freqs&predictor prefix=frequency;
id &response; var Freqdisplay;
run;
proc sql;
select label into :label from sashelp.vcolumn
where libname="%upcase(&directory)" and memname="%upcase(&dataset)" and
upcase(name)="%upcase(&predictor)";
quit;
data Freqs&predictor; length predictor $98; drop _Name_; set Freqs&predictor;
predictor = upcase("&predictor");
characteristic="&label";

```

```

level=.; /* avoids an error from proc append */
run;
/* significance test */
proc npar1way data=&directory..&dataset wilcoxon;
var &predictor; class &response;
output out=p&predictor wilcoxon;
run;
data p&predictor(keep=predictor P_PCHI); length predictor $98; set
p&predictor(rename=(P_KW=P_PCHI));
predictor=upcase("&predictor");
run;
/* merge summary and significance test */
data final&predictor;
merge Freqs&predictor p&predictor; by predictor;
run;
proc append data=final&predictor base=final force; run;
proc datasets nolist;
delete summarya&predictor summaryb&predictor summary&predictor freqs&predictor
p&predictor final&predictor;
quit;
%end;
%end;

/**** CREATE COLUMN LABEL MACRO VARIABLES ****/
ods output OneWayFreqs=OneWay&response;
proc freq data=&directory..&dataset;
tables &response;
run;
ods output close;
proc sql;
select &response, &response, put(&response,%sysfunc(cat(&response_format.,.))),
cats(put(Frequency,comma8.))
into :response_levels separated by ' ',:resplvl1-:resplvl999,:resplbl1-
:resplbl999,:respcum1-:respcum999
from OneWay&response;
quit;
proc sql;
select cats(put(max(CumFrequency),comma8.)) into :overall_count from OneWay&response;
quit;
proc datasets nolist; delete OneWay&response; quit;

/**** CREATE FINAL REPORT ****/
ods listing; title; footnote; ods listing close;
ods rtf;
%let st=style(column)=[just=center cellwidth=2.8 cm vjust=bottom font_size=8.5 pt]
style(header)=[just=center font_size=8.5 pt];
options orientation=portrait missing=' ';
proc report data=final nowd style=[cellpadding=6 font_size=8.5 pt rules=none];
column (characteristic frequency_99
%do j=1 %to %sysfunc(countw(&response_levels)); frequency&&resplvl&j %end;
P_PCHI);
define characteristic / display " " ;
define frequency_99 / display "{Overall\line (N=&overall_count)}" &st ;
%do k=1 %to %sysfunc(countw(&response_levels));
define frequency&&resplvl&k / display "&&resplbl&k\line (N=&&respcum&k)}" &st;
%end;
define P_PCHI / display "{p-value}" &st format=pvalue_best.;
run;
ods rtf close;

%mend table1;

```