Customize comparison tables for clinical studies

Polina Kukhareva, University of Utah, Salt Lake City, UT

Kathy Roggenkamp, Collaborative Studies Coordinating Center, UNC, Chapel Hill, NC

ABSTRACT

Comparisons between two or more groups are very common in clinical studies. This paper describes a two-step strategy to create customized tables containing correct comparative p-values and descriptive statistics faster and easier. At the first step, you use a SAS macro to create a data set and an RTF file for the descriptive statistics and p-values. This macro has three advantages over performing the estimations one by one. First, the macro makes it easier to compare a long list of baseline characteristics. Secondly, it allows variables to be added or deleted from the list. Thirdly, it works with comparisons of two or more than two groups. The macro can save hours of code-writing time for a programmer. At the second step, you can change variable labels, titles, footnotes, or the style of the report, by using the DATA step and PROC REPORT. Program code for post processing as well as the macro code is included. In addition, this two-step strategy facilitates producing tables that comply with common journal requirements.

INTRODUCTION

Almost any statistical analysis starts with producing descriptive characteristics. Producing the correct p-values and descriptive statistics can be time-consuming, especially when they are needed for several tables containing a lot of variables of different types (categorical, continuous, and countable). This article describes a two-step solution which makes producing correct reports fast and easy.

Working as a statistical programmer for a large clinical study I often need to write and run code for creating multiple comparison tables of baseline characteristics. It’s so easy to make a mistake and forget to add or delete one variable from a long list, or use a wrong label, or perform a wrong statistical test. So over time I developed some techniques which make producing those tables easier by enabling me to run all the calculations in one macro call, and then choose some lines to create a customized table.

My experience comes from a randomized clinical trial but the macro is probably even more useful for a clinical study where participants are not randomized and the primary analysis more often requires controlling for lots of baseline covariates of interest and assessment for confounding. Also, this two-step strategy can be applied to any other type of study, not necessarily belonging to the biomedical domain.

Objectives

1. Describe a macro which:
   - creates a table of descriptive statistics of sample characteristics across the group categories
   - estimates correct p-values for categorical, countable, and continuous variables
   - prints this table into an RTF file and into a new SAS data set
2. Describe post-processing code which
   - creates a report containing only specific lines
   - uses new format for p-values, new report style, new titles and footnotes
   - uses in-line formatting to change fonts in the labels

INPUT DATA SET

All the tables presented in this paper are generated using a simulated data set. This data set includes response variable event (events coded as 1, censored observations coded as 0), follow-up variable time, five continuous predictor variables (age, systolic_bp, diastolic_bp, ldl, bmi) and five categorical predictors (diabetes, smoking, sex, treatment group, activity). Activity has three levels (‘low’, ‘high’, ‘average’) and all other categorical variables have two levels (‘0’, ‘1’). The data set is intended to represent a small data set from a cardiovascular drug study with time following an exponential distribution.. The data set contains 3000 observations, and some values of all variables are randomly assigned to missing.

STEP 1: RUNNING THE MACRO

The macro was developed in SAS 9.3 but works equally well in SAS 9.2. The following procedures are used in the macro: SQL, DATASETS, FORMAT, FREQ, TRANSPOSE, APPEND, NPAR1WAY, UNIVARIATE, ANOVA, REPORT, and SORT.

MACRO PARAMETERS

The macro has nine parameters which are specified in Table 1. Out of those nine parameters four are required. The other five have default values and thus only need to be defined in the macro call if the user wants to overwrite their default values.

Parameter	Description	Example
Required Parameters		
_DATA_IN	Name of the data set containing initial data (must exist)	rq.simcox
_DATA_OUT	data set containing results (will be created)	data_out1
_GROUP	by variable	treatment
_CHARACTERISTICS	List of variables to be included in a table separated by blanks	age systolic_bp diastolic_bp ldl bmi diabetes smoking sex
Optional Parameters		
CATEGORICAL= no_categorical_variables	List of ALL the categorical variables separated by blanks	diabetes smoking sex treatment activity
COUNTABLE= no_countable_variables	List of ALL the variables for which we estimate median and IQR	ldl
_FOOTNOTE= %str(&sysdate, &systime -- produced by macro Compare_baseline_characteristics)	Footnote which appears in the RTF file	%str(&sysdate, &systime -- produced by macro Compare_baseline_characteristics)
_TITLE1= Compare_baseline_characteristics Macro	Title which appears in the RTF file	Compare_baseline_characteristics Macro
_NUMBER= bc_macro_1	Characters to be included in the name of output RTF file	bc_macro_1

Table 1. Macro Parameters

CALLING THE MACRO

Below you can see two examples of calling the macro.

Example 1

The first example is a comparison between two treatment groups. In this example the by variable has two categories: treatment A and treatment B. Since those categories were coded as 0 and 1, 0 and 1 are presented in the heading of the table. The macro also produces a number of observations in each group and overall (specifically, the number of observations with a non-missing value for the _GROUP parameter variable). The *variable label* column contains the variable label or variable values (for categorical variables). The *variable name* column contains upper-cased variable names.

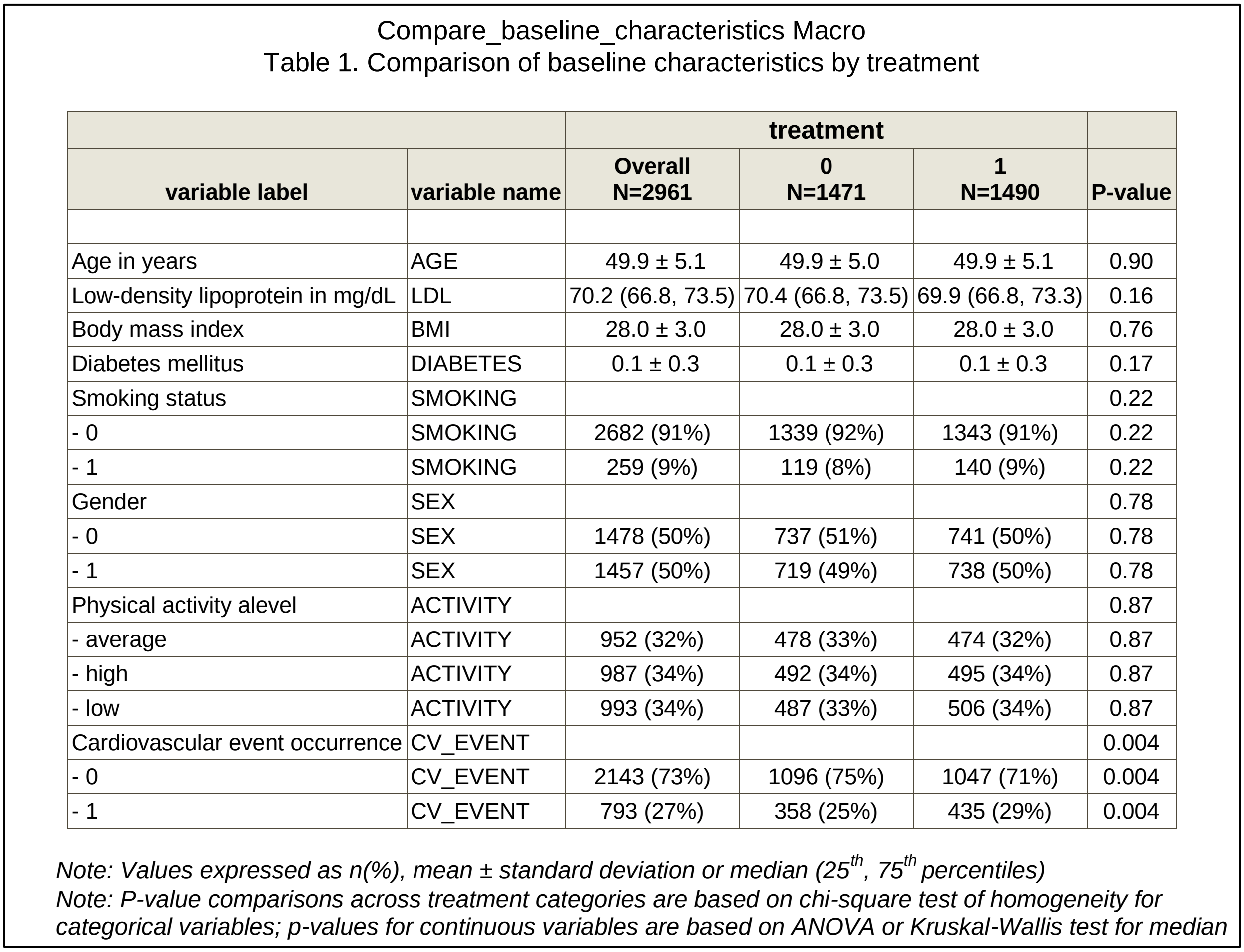
Different descriptive statistics and comparison tests need to be run for continuous and categorical variables. The macro runs different statistics for those variables mentioned in the _CATEGORICAL (categorical characteristics) and _COUNTABLE parameters from those produced for continuous predictors.

The *Overall* column contains descriptive characteristics for the whole data set. The next two columns correspond to the two levels of treatment. P-value comparisons across treatment groups for categorical variables are based on the chi-square test of homogeneity; p-values for continuous variables are based on the ANOVA or Kruskal-Wallis median test. P-value comparisons across activity categories are based on the chi-square test of homogeneity for categorical variables; p-values for continuous variables are based on the ANOVA or Kruskal-Wallis median test.

Note that the *Overall* column contains only those observations which do not have a missing value for treatment (this example's _GROUP variable). If you need to have an overall column which includes observations with missing treatment values as well, just comment out line `if missing (&_GROUP) then delete;` in the macro code.

```
%Compare_baseline_characteristics (_DATA_IN=rq.simcox, _DATA_OUT=data_out1,
_group=treatment, _CHARACTERISTICS= age ldl bmi diabetes smoking sex activity
CV_event,_CATEGORICAL=smoking sex treatment activity CV_event, _countable=ldl,
_NUMBER=bc_macro_1)
```

This macro call produces both a new SAS data set and RTF table. RTF table is presented in the Output 1 below.



Compare_baseline_characteristics Macro
Table 1. Comparison of baseline characteristics by treatment

		treatment			
variable label	**variable name**	**Overall N=2961**	**0 N=1471**	**1 N=1490**	**P-value**
Age in years	AGE	49.9 ± 5.1	49.9 ± 5.0	49.9 ± 5.1	0.90
Low-density lipoprotein in mg/dL	LDL	70.2 (66.8, 73.5)	70.4 (66.8, 73.5)	69.9 (66.8, 73.3)	0.16
Body mass index	BMI	28.0 ± 3.0	28.0 ± 3.0	28.0 ± 3.0	0.76
Diabetes mellitus	DIABETES	0.1 ± 0.3	0.1 ± 0.3	0.1 ± 0.3	0.17
Smoking status	SMOKING				0.22
- 0	SMOKING	2682 (91%)	1339 (92%)	1343 (91%)	0.22
- 1	SMOKING	259 (9%)	119 (8%)	140 (9%)	0.22
Gender	SEX				0.78
- 0	SEX	1478 (50%)	737 (51%)	741 (50%)	0.78
- 1	SEX	1457 (50%)	719 (49%)	738 (50%)	0.78
Physical activity alevel	ACTIVITY				0.87
- average	ACTIVITY	952 (32%)	478 (33%)	474 (32%)	0.87
- high	ACTIVITY	987 (34%)	492 (34%)	495 (34%)	0.87
- low	ACTIVITY	993 (34%)	487 (33%)	506 (34%)	0.87
Cardiovascular event occurrence	CV_EVENT				0.004
- 0	CV_EVENT	2143 (73%)	1096 (75%)	1047 (71%)	0.004
- 1	CV_EVENT	793 (27%)	358 (25%)	435 (29%)	0.004

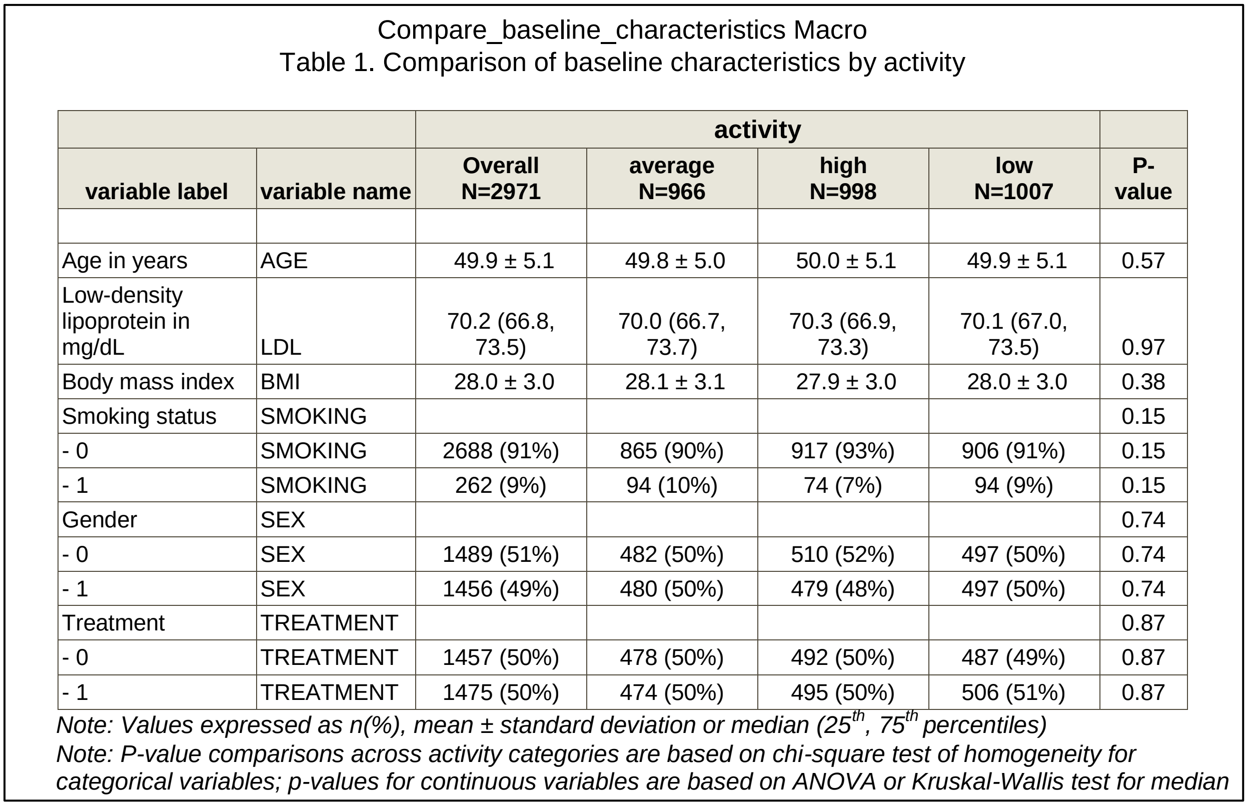
Note: Values expressed as n(%), mean ± standard deviation or median (25th, 75th percentiles)
Note: P-value comparisons across treatment categories are based on chi-square test of homogeneity for categorical variables; p-values for continuous variables are based on ANOVA or Kruskal-Wallis test for median

Output 1. Raw Table from Example 1 Produced by Macro Call

Example 2

The second example shows a comparison by an activity variable which has three groups. The resulting table is presented in Output 2.

```
%Compare_baseline_characteristics (_DATA_IN=rq.simcox, _DATA_OUT=data_out2,
_group=activity,_CHARACTERISTICS= age ldl bmi smoking sex
treatment,_categorical=smoking sex treatment activity, _countable=ldl,
_NUMBER=bc_macro_2)
```



Compare_baseline_characteristics Macro
Table 1. Comparison of baseline characteristics by activity

		activity				
variable label	**variable name**	**Overall N=2971**	**average N=966**	**high N=998**	**low N=1007**	**P-value**
Age in years	AGE	49.9 ± 5.1	49.8 ± 5.0	50.0 ± 5.1	49.9 ± 5.1	0.57
Low-density lipoprotein in mg/dL	LDL	70.2 (66.8, 73.5)	70.0 (66.7, 73.7)	70.3 (66.9, 73.3)	70.1 (67.0, 73.5)	0.97
Body mass index	BMI	28.0 ± 3.0	28.1 ± 3.1	27.9 ± 3.0	28.0 ± 3.0	0.38
Smoking status	SMOKING					0.15
- 0	SMOKING	2688 (91%)	865 (90%)	917 (93%)	906 (91%)	0.15
- 1	SMOKING	262 (9%)	94 (10%)	74 (7%)	94 (9%)	0.15
Gender	SEX					0.74
- 0	SEX	1489 (51%)	482 (50%)	510 (52%)	497 (50%)	0.74
- 1	SEX	1456 (49%)	480 (50%)	479 (48%)	497 (50%)	0.74
Treatment	TREATMENT					0.87
- 0	TREATMENT	1457 (50%)	478 (50%)	492 (50%)	487 (49%)	0.87
- 1	TREATMENT	1475 (50%)	474 (50%)	495 (50%)	506 (51%)	0.87

Note: Values expressed as n(%), mean ± standard deviation or median (25th, 75th percentiles)
Note: P-value comparisons across activity categories are based on chi-square test of homogeneity for categorical variables; p-values for continuous variables are based on ANOVA or Kruskal-Wallis test for median

Output 2. Raw Table from Example 2 Produced by Macro Call

STEP 2: PRODUCING A CUSTOMIZED REPORT

As a second step, you can change variable labels, titles, footnotes, row order, or the style of the report, using the DATA step and PROC REPORT to produce any report you like from the data set produced by the macro. This code is a little long but it isn't that hard to write. I usually use a copy of the RTF table produced by the macro to draft it. First I delete all columns except for variable name and variable label. Second, I delete rows which I don't need. Third, I add some additional columns to that table in Microsoft Word (Columns 1, 2, and 3), and then I copy this table to the SAS editor and delete extra blanks. The only column which I need to edit is column 3, where I put new labels and the desired order. The RTF table from the Output 1 above has been modified in this way to produce Table 2 below.

Column 1	variable name	Column 2	variable label	Column 3
`if variable="`	`AGE`	`" and label="`	`Age in years`	`" then do; characteristic="{\i \ul Baseline Characteristics\line \line \ul0 \i0 &c Age (yr)}"; order=`**1**`; end;`
`if variable="`	`LDL`	`" and label="`	`Low-density lipoprotein in mg/dL`	`" then do; characteristic="{&c LDL cholesterol (mg/dL), median (25\super th}{, 75\super th }{)}"; order=`**9**`; end;`
`if variable="`	`BMI`	`" and label="`	`Body mass index`	`" then do; characteristic="{&c Body mass index (kg/m)}"; order=`**2**`; end;`
`if variable="`	`SMOKING`	`" and label="`	`- 1`	`" then do; characteristic="&c Currently smoking"; order=`**3**`; end;`
`if variable="`	`SEX`	`" and label="`	`- 0`	`" then do; characteristic="&c Women"; order=`**4**`; end;`
`if variable="`	`ACTIVITY`	`" and label="`	`Physical activity alevel`	`" then do; characteristic="&c Exercise level" order=`**5**`; end;`
`if variable="`	`ACTIVITY`	`" and label="`	`- average`	`" then do; characteristic="{&c&c&c Average}"; order=`**6**`; pvalue=.;  end;`

if variable="	ACTIVITY	" and label="	- high	" then do; characteristic="{&c&c&c High}"; order=**7**; pvalue=.; end;
if variable="	ACTIVITY	" and label="	- low	" then do; characteristic="{&c&c&c Low}"; order=**8**; pvalue=.; end;
if variable="	CV_EVENT	" and label="	- 1	" then do; characteristic"{\i \ul Outcome\line \line \ul0 \i0 &c Cardiovascular Event Occurrence}"; order=**10**; end;

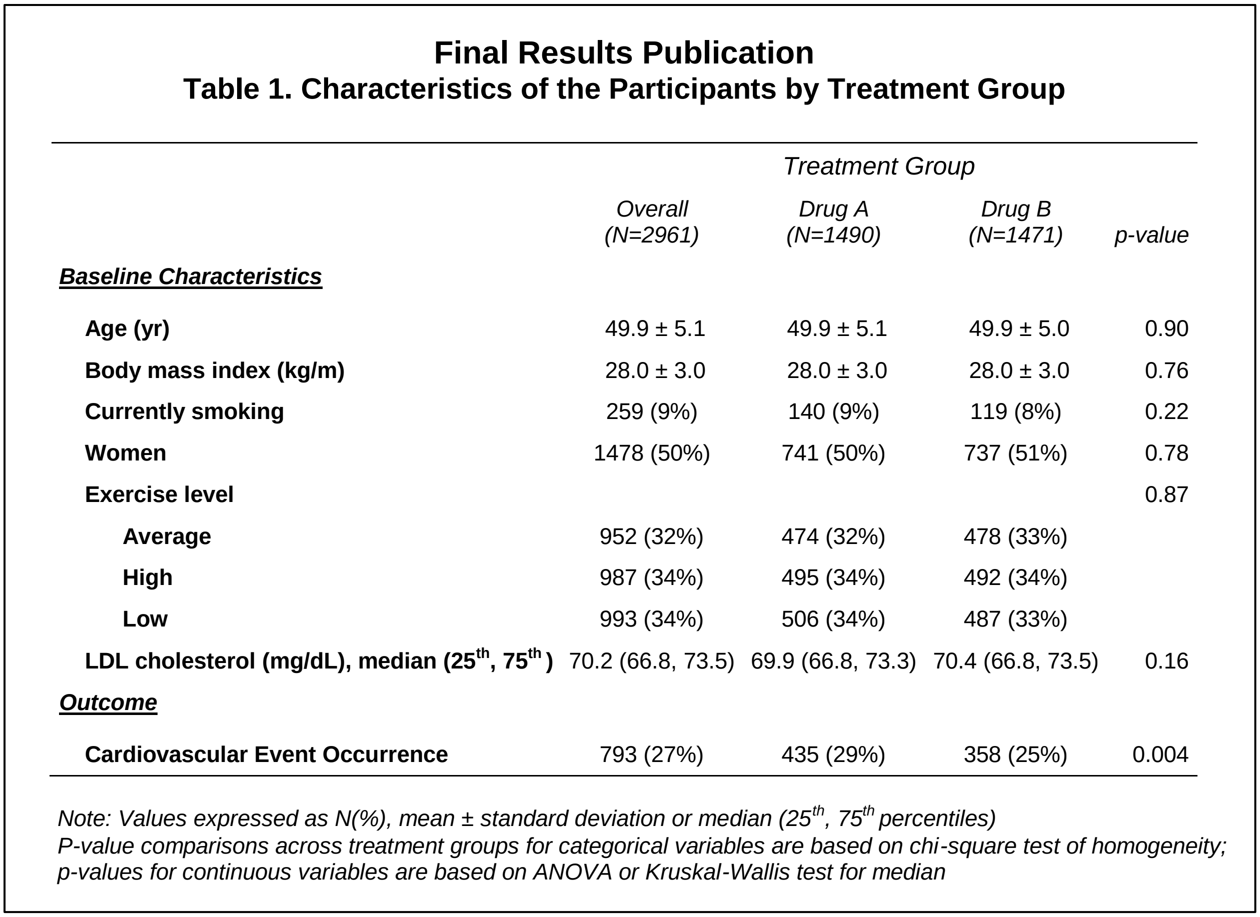
Table 2. Supporting Table which Helps in Creating the SAS Code

The following code based on Table 2 can be used to produce a customized table. Note that I used PROC FORMAT to create a new format for representing p-values. You can change it any way you need according to journal specifications. Also note that I used inline formatting which helps with superscripting, italic text, underlining, and skipping to the next line. For example, \i makes the subsequent text appear in italics, and \line inserts a line break. Using &c helped me to align labels the way I needed. With the order variable I changed the order of baseline characteristics presented in Output 3.

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

proc format;
   value pvalue2_best      0-<0.001='<0.001'   0.001-<0.005=[5.3]
                           0.005-<0.045= [5.2] 0.045-<0.055=[5.3] other=[5.2];
run;

data display;
   set data_out1;
   length characteristic $200;
   if variable="AGE" and label="Age in years" then do;
      characteristic="{\i \ul Baseline Characteristics\line \line \ul0 \i0 &c Age
(yr)}";
      order=1; end;
   if variable="LDL" and label="Low-density lipoprotein in mg/dL" then do;
      characteristic="{&c LDL cholesterol (mg/dL), median (25\super th}{, 75\super th
}{)}"; order=9; end;
   if variable="BMI" and label="Body mass index" then do;
      characteristic="{&c Body mass index (kg/m)}"; order=2; end;
   if variable="SMOKING" and label="- 1" then do;
      characteristic="&c Currently smoking"; order=3; end;
   if variable="SEX" and label="- 0" then do; characteristic="&c Women"; order=4; end;
   if variable="ACTIVITY" and label="Physical activity alevel" then do;
      characteristic="&c Exercise level"; order=5; end;
   if variable="ACTIVITY" and label="- average" then do;
      characteristic="{&c&c&c Average}"; order=6; pvalue=.; end;
   if variable="ACTIVITY" and label="- high" then do;
      characteristic="{&c&c&c High}"; order=7; pvalue=.;  end;
   if variable="ACTIVITY" and label="- low" then do;
      characteristic="{&c&c&c Low}"; order=8; pvalue=.;  end;
   if variable="CV_EVENT" and label="- 1" then do;
      characteristic="{\i \ul Outcome\line \line \ul0 \i0 &c Cardiovascular Event
Occurrence}"; order=10; end;
   if missing (order) then delete;
run;

ODS RTF FILE="&odsdir.\customised_table.RTF" style=journal bodytitle;
ods listing; title; footnote; ods listing close;

title1 J=center height=12pt font='ARIAL' bold "Final Results Publication";
title2  J=center height=11pt bold font='ARIAL' "{Table 1. Characteristics of the
Participants by Treatment Group}";
footnote1 J=left height=8.5pt font='ARIAL'
"{Note: Values expressed as N(%), mean ± standard deviation or median (25\super th}{,
75\super th }{percentiles)}" ;
```

```
footnote2 J=left height=8.5pt font='ARIAL'
"P-value comparisons across treatment groups for categorical variables are based on
chi-square test of homogeneity; p-values for continuous variables are based on ANOVA
or Kruskal-Wallis test for median" ;
Footnote3 J=left height=8.5pt font='ARIAL'" ";
Footnote4 J=right height=7pt font='ARIAL'
   "&sysdate, &systime -- Baseline Characteristics Macro";
%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];

proc report data=display nowd style=[cellpadding=6 font_size=8.5 pt rules=none];
   column order characteristic('Treatment Group' column_overall column_2 column_1
pvalue);
   define order / order noprint;
   define characteristic / display " "
      style=[just=left cellwidth=9.0 cm font_weight=bold font_size=8.5 pt];
   define column_2 / display "{Drug A\line (N=&count_2)}" &st ;
   define column_1 / display "{Drug B\line (N=&count_1)}" &st ;
   define column_overall / display "{Overall\line (N=&count_overall)}" &st ;
   define pvalue / display "{p-value}" format=pvalue2_best.
      style(column)=[just=right cellwidth=2 cm vjust=bottom font_size=8.5 pt]
      style(header)=[just=right cellwidth=2 cm font_size=8.5 pt] ;
run;

ods rtf close; ods listing;
```

Output 3 shows the resulting customized table.



Final Results Publication

Table 1. Characteristics of the Participants by Treatment Group

	Treatment Group			
	Overall *(N=2961)*	*Drug A* *(N=1490)*	*Drug B* *(N=1471)*	*p-value*
Baseline Characteristics				
Age (yr)	49.9 ± 5.1	49.9 ± 5.1	49.9 ± 5.0	0.90
Body mass index (kg/m)	28.0 ± 3.0	28.0 ± 3.0	28.0 ± 3.0	0.76
Currently smoking	259 (9%)	140 (9%)	119 (8%)	0.22
Women	1478 (50%)	741 (50%)	737 (51%)	0.78
Exercise level				0.87
Average	952 (32%)	474 (32%)	478 (33%)	
High	987 (34%)	495 (34%)	492 (34%)	
Low	993 (34%)	506 (34%)	487 (33%)	
LDL cholesterol (mg/dL), median (25th, 75th)	70.2 (66.8, 73.5)	69.9 (66.8, 73.3)	70.4 (66.8, 73.5)	0.16
Outcome				
Cardiovascular Event Occurrence	793 (27%)	435 (29%)	358 (25%)	0.004

Note: Values expressed as N(%), mean ± standard deviation or median (25th, 75th percentiles)
P-value comparisons across treatment groups for categorical variables are based on chi-square test of homogeneity; p-values for continuous variables are based on ANOVA or Kruskal-Wallis test for median

Output 3. Customized Table

CONCLUSION

This two-step strategy allows producing comparison tables that comply with journal requirements. The macro can help to save hours of work for a programmer performing statistical analysis. Also this macro can save time for a biostatistician writing a request. It is so much easier to say 'Run the baseline characteristic macro' than to write out everything that's needed. This time savings can be multiplied with several clinical studies. It can minimize errors which can be made specifying baseline tables. The customization ideas might be helpful if slight changes are needed in existing reports.

ACKNOWLEDGMENTS

Thanks to our colleagues at the Collaborative Studies Coordinating Center: Myra Carpenter, Stephen Campbell, and Kwanhye Jung. This work was supported by the Folic Acid for Vascular Outcome Reduction in Transplantation (FAVORIT) trial cooperative agreement U01 DK61700 from the National Institute of Diabetes and Digestive and Kidney Diseases, National Institutes of Health. The content is solely the responsibility of the authors.

CONTACT INFORMATION

Your comments and questions are valued and encouraged.

Name: Polina Kukhareva
Enterprise: Department of Biomedical Informatics, University of Utah
Address: 651 Arapeen Dr., Suite 208, Salt Lake City, Utah, 84108
Work Phone: 801-587-8001
E-mail: p.kukhareva@alumni.unc.edu,
Web: www.linkedin.com/pub/polina-kukhareva/29/42a/7a1

MACRO CODE

```
*  MACRO:        Compare_baseline_characteristics
*  DESCRIPTION:  Compares baseline characteristics by a specified variable
*  SOURCE:       CSCC, UNC at Chapel Hill
*  PROGRAMMER:   Polina Kukhareva
*  DATE:         05/13/2013
*  LANGUAGE:     SAS VERSION 9.3
*************************************************************;
%macro Compare_baseline_characteristics(
_DATA_IN=/*Name of the data set containing initial data, e.g. rq.simcox */,
_DATA_OUT=/*data set containing results, e.g. data_out1*/,
_GROUP=/*by variable, e.g. treatment*/,
_CHARACTERISTICS=/*List of variables to be included in a table separated by blanks*/,
_CATEGORICAL=_no_categorical_variables/*List of ALL the categorical variables
separated by blanks*/,
_COUNTABLE=_ no_countable_variables/*List of ALL the variables for which we estimate
median and IQR*/,
_FOOTNOTE=%str(&sysdate, &systime -- produced by macro
Compare_baseline_characteristics) /*Footnote which appears in the rtf file */,
_TITLE1=Compare_baseline_characteristics Macro/*Title which appears in the rtf file*/,
_NUMBER= bc_macro_1/*Characters which will appear in the name of rtf file, e.g.
bc_macro_1*/)/ minoperator;

options nodate mprint pageno=1 mergenoby=warn MISSING=' ' validvarname=upcase;
%let _CHARACTERISTICS=%upcase(&_CHARACTERISTICS);
%put _CHARACTERISTICS= &_CHARACTERISTICS;
%let _CATEGORICAL=%upcase(&_CATEGORICAL);
%put _CATEGORICAL= &_CATEGORICAL;
%let _COUNTABLE=%upcase(&_COUNTABLE);
%global count_1 count_2 count_3 count_4 count_5 count_6 count_7 count_8 count_overall;
proc format;
   value pvalue_best
      0-<0.1=[pvalue5.3]
      Other=[5.2] ;
run;

/*Producing a work data set*/
data baseline_characteristics_ds;
   set &_DATA_IN;
      length categorical_group $100;
      if Vtype(&_GROUP)='C' then categorical_group=&_GROUP;
         else categorical_group=strip(input(&_GROUP, best12.));
      if missing (&_GROUP) then delete;
run;

proc sort data=baseline_characteristics_ds;
   by categorical_group;
run;

proc sql;
   select distinct categorical_group into :distinct_groups separated by '~'
      from baseline_characteristics_ds;
   quit;
%let number_of_distinct_groups= %eval(%sysfunc(countw(%str(&distinct_groups),~)));
%do i=1 %to &number_of_distinct_groups;
   %let categorical_group_&i=%scan(&distinct_groups,&i,~);
      proc sql;
      select count (*) into :count_&i from baseline_characteristics_ds
             where categorical_group="&&categorical_group_&i";
       quit;
       %let count_&i=&&count_&i;
```

```
%end;
proc sql;
   select count (*) into :count_overall from baseline_characteristics_ds;
quit;
%let count_overall=&count_overall;
/*Creating an empty data set to append some observations later*/
data table2;
   length label $ 100 variable $ 40 %do i=1 %to &number_of_distinct_groups;
       column_&i $ 200 %end; column_overall $200 pvalue 8;
run;

/*We are iterating through all the predictors in given order to compare their values
between excluded and included data sets*/
%do  all_count=1 %to %sysfunc(countw(&_CHARACTERISTICS));
   %let CHECK_VAR=%scan(&_CHARACTERISTICS, &all_count,%str( ));
   %let CHECK_VAR=%UNQUOTE(&CHECK_VAR);
       /*We calculate number, percentage and p-value using chi-square test for
categorical predictors*/
   %if &CHECK_VAR in &_categorical %then %do;

      /*getting p-values*/
      proc freq data=baseline_characteristics_ds;
         table categorical_group*&CHECK_VAR/chisq;
         output out=p pchi;
      run;
      %if (%sysfunc(exist(work.p)))=0 %then %do;
         data p;
            length p_pchi pvalue 8.;
         run;
      %end;
      data p;
         set p(keep=p_pchi rename=(p_pchi=pvalue));
      run;
      /*getting percentages*/
      %do i=1 %to &number_of_distinct_groups;
         proc sql;
            create table part1_&i as
            select a.&CHECK_VAR as label1,
            strip(put(count(a.&CHECK_VAR),8.0))||' ('||
            strip(put(count(a.&CHECK_VAR)/Subtotal,percent8.0))||')' as column_&i
            from baseline_characteristics_ds as a,
            (select count(&CHECK_VAR) as Subtotal from baseline_characteristics_ds
            where categorical_group="&&categorical_group_&i")
            where ^missing(&CHECK_VAR) and categorical_group="&&categorical_group_&i"
            group by a.&CHECK_VAR ;
         quit;
      %end;
      proc sql;
         create table part1_overall as
            select a.&CHECK_VAR as label1, strip(put(count(a.&CHECK_VAR),8.0))||' ('||
            strip(put(count(a.&CHECK_VAR)/Subtotal,percent8.0))||')' as column_overall
            from baseline_characteristics_ds as a,
            (select count(&CHECK_VAR) as Subtotal from baseline_characteristics_ds)
            where ^missing(&CHECK_VAR)
            group by a.&CHECK_VAR ;
      quit;
      data part1 (drop=label1);
         length label $100;
         merge %do i=1 %to &number_of_distinct_groups; part1_&i %end; part1_overall;
         by label1;
         if Vtype(label1)='C' then label=label1;
         else label=put(label1, 8.0);
      run;
      data part1;
         set part1;
         length label $100;
         label='- '||strip(label);
         variable="&CHECK_VAR";
      run;
      proc sql; create table part1 as select * from part1, p; quit;
      /*getting label*/
      proc TRANSPOSE DATA=baseline_characteristics_ds (OBS=1 KEEP=&CHECK_VAR)
OUT=VARLABL;
         var &CHECK_VAR;
      run;
      /* checking existence of the variable label */
      data _null_;
           dsid=open('VARLABL');
           check_VARLABL=varnum(dsid,'_Label_');
            call symput('check_label',put(check_VARLABL,best.));
      run;
      data VARLABL;
         length _label_ $40;
         set VARLABL;
          %if &check_label=0 %then %do; _Label_=' '; %end;
      run;
      /* merging p-values and labels */
      data part2;
         set p;
         set VARLABL (keep=_name_  _Label_ rename=(_Label_=label _name_=variable));
      run;

      data add; set part2 part1; run;

      proc append BASE=table2 DATA=add force; run;

   %end;
   /*We calculate median, IQR and p-value using Kruskal-Wallis test for median for not
normally distributed continuous predictors*/
   %else %if &CHECK_VAR in &_countable %then %do;
      /*getting p-value*/
      proc npar1way data=baseline_characteristics_ds wilcoxon;
         var &CHECK_VAR;
         class categorical_group;
         output out=p Wilcoxon;
      run;
      /*getting median and IQR*/
      proc univariate data=baseline_characteristics_ds noprint;
         var &CHECK_VAR;
         output out=IQR pctlpts= 25 50 75 pctlpre=&CHECK_VAR.;
         by categorical_group;
      run;
      proc univariate data=baseline_characteristics_ds noprint;
         var &CHECK_VAR;
         output out=IQR_overall pctlpts= 25 50 75 pctlpre=&CHECK_VAR.;
      run;
      data IQR;
         format tval $50.;
         set iqr (where =(^missing(categorical_group))) IQR_overall;
         length IQR_group $100;
         tval="{"||(strip(put(&CHECK_VAR.50,5.1)))||' ('
         ||strip(put(&CHECK_VAR.25,5.1))||', '||strip(put(&CHECK_VAR.75,5.1))||')}';
         drop &CHECK_VAR.50 &CHECK_VAR.25 &CHECK_VAR.75;
         %do i=1 %to &number_of_distinct_groups;
            if categorical_group="&&categorical_group_&i" then IQR_group="column_&i";
         %end;
         if missing(IQR_group) then IQR_group="column_overall";
      run;
      /*getting label*/
      proc transpose data=IQR out=median_p_trans; id IQR_group; var tval; run;

      proc transpose DATA=baseline_characteristics_ds(OBS=1 KEEP=&CHECK_VAR)
OUT=VARLABL;
      run;
      /* checking existence of the variable label */
      data _null_;
         dsid=open('VARLABL');
         check_VARLABL=varnum(dsid,'_Label_');
         call symput('check_label',put(check_VARLABL,best.));
      run;
      data VARLABL;
         length _label_ $40;
         set VARLABL;
         %if &check_label=0 %then %do; _Label_=' '; %end;
      run;

      data add;
         set median_p_trans
            (keep=%do i=1 %to &number_of_distinct_groups; column_&i %end;
column_overall);
         set p (keep=P_KW rename=(P_KW=pvalue));
         set VARLABL (keep=_name_ _Label_ rename=(_Label_=label _name_=variable));
      run;
      proc append BASE=table2 DATA=add force;
      run;
   %end;
   /*We calculate mean, standard deviation and p-value using T-test for continuous
predictors*/
   %else %do;
   /*getting mean and std*/
      %do i=1 %to &number_of_distinct_groups;
         proc sql;
            create table part1_&i
            as select catx(' ','{', put(mean(&CHECK_VAR), 8.1),
            ' \u0177\~ ',put(sqrt(var(&CHECK_VAR)),8.1),'}') as column_&i
            from baseline_characteristics_ds
            where categorical_group="&&categorical_group_&i";
            quit;
      %end;
      proc sql;
         create table part1_overall
         as select catx(' ','{', put(mean(&CHECK_VAR), 8.1),' \u0177\~ ',
         put(sqrt(var(&CHECK_VAR)),8.1),'}') as column_overall
         from baseline_characteristics_ds;
      quit;
      /* getting p-value*/
      ods output OverallANOVA=p(keep= dependent source probf where=(source='Model')
         rename=(probf=pvalue dependent=variable));
      proc anova data=baseline_characteristics_ds;
         class categorical_group;
            model  &CHECK_VAR=categorical_group;
      run; quit;
      ods output close;
      /*getting label*/
      proc transpose DATA=baseline_characteristics_ds (OBS=1 KEEP=&CHECK_VAR)
OUT=VARLABL;
      run;
      /* checking existence of the variable label */
      data _null_;
```

```
         dsid=open('VARLABL');
         check_VARLABL=varnum(dsid,'_Label_');
         call symput('check_label',put(check_VARLABL,best.));
      run;
      data VARLABL;
         length _label_ $40;
         set VARLABL;
         %if &check_label=0 %then %do; _Label_=' '; %end;
      run;
      data add;
         %do i=1 %to &number_of_distinct_groups; set part1_&i;%end;
         set part1_overall;
         set p (keep=pvalue);
         set VARLABL (keep=_name_ _Label_ rename=(_Label_=label _name_=variable));
      run;
      proc append BASE=table2 DATA=add force; run;
   %end;
   proc datasets lib=work memtype=data; delete p ; run; quit;
%end;
data &_DATA_OUT;
   set table2;
run;

/*Printing table 1 in rtf destination*/
title1 j=center height=12pt font="Times Roman" "&_TITLE1";
title2 j=center height=12pt font="Times Roman" "Table 1. Comparison of baseline
characteristics by &_GROUP";
footnote1 J=left height=9pt font="TIMES ROMAN" "{Note: Values expressed as n(%), mean
± standard deviation or median (25\super th}{, 75\super th }{percentiles)}";
footnote2 J=left height=9pt font="TIMES ROMAN"
"Note: P-value comparisons across &_GROUP categories are based on chi-square test of
homogeneity for categorical variables; p-values for continuous variables are based on
ANOVA or Kruskal-Wallis test for median";
footnote3 J=right height=9pt font="TIMES ROMAN" &_FOOTNOTE;

ods listing close;
ods rtf file="&_NUMBER._&_Group..rtf" style=analysis bodytitle;
ods rtf startpage=NO;
%let st=style(column)=[just=center vjust=bottom font_size=8.5 pt]
        style(header)=[just=center font_size=8.5 pt];
proc report data=table2 nowd ;
   column label variable  ("&_GROUP" column_overall
   %do i=1 %to &number_of_distinct_groups; column_&i %end;)  pvalue;
   define label / 'variable label' display
      style(column)=[just=left vjust=bottom font_size=8.5 pt]
      style(header)=[just=center font_size=8.5 pt];
   define variable / 'variable name' display
      style(column)=[just=left vjust=bottom font_size=8.5 pt]
      style(header)=[just=center font_size=8.5 pt];
   define column_overall / "Overall / N=&count_overall" display &st;
   %do i=1 %to &number_of_distinct_groups;
      define column_&i / "&&categorical_group_&i / N=&&count_&i" display &st;
   %end;
   define pvalue / 'P-value' display format=pvalue_best. &st;
run;
ods rtf exclude all;

proc datasets lib=work memtype=data; delete table2 baseline_characteristics_ds ;
run; quit;
ods rtf close;
ods listing;
footnote; title;
%mend Compare_baseline_characteristics;
```