

Paper 096-2010

Automatically Run Fisher's Exact Test When the Chi-square Test Might Not Be Valid

Wei Xu, Boston Scientific, Marlborough, Massachusetts

ABSTRACT

Have you ever think about how to automatically perform Fisher's exact test whenever a WARNING message for the Chi-Square test is issued? If PROC FREQ issues a WARNING message that the Chi-Square test might not be valid due to a large number of cells with small expected frequency, you will need to run Fisher's exact test instead of Chi-square test per your analysis requests.

This paper will describe a simple, straightforward way to use a %RUN_FISHERS macro automatically performing Fisher's exact test when the Chi-square test might not be valid. When there are exactly two rows and two columns and if any of the expected cell frequencies are less than five, the macro will automatically use Fisher's exact test. The macro %RUN_FISHERS is only needed when the table that is larger than 2x2 since the CHISQ option in the TABLE statement of PROC FREQ provides both the chi-square test and Fisher's exact test for 2x2 tables.

THE SETUP

You will need a few things to get this set up.

1. Create the project location of the %RUN_FISHERS macro definition file on your system.
2. In your SAS® program or in the SAS editor window, specify the statement to define the %RUN_FISHERS macro and make it available for use.
3. Following the statement, you need to call the %RUN_FISHERS macro.

THE MACRO %RUN_FISHERS

Specify this statement to define the %RUN_FISHERS macro and make it available for use:

```

/*****
%include "<location of your file containing the RUN_FISHERS macro>";
*****/

```

The following parameters are required:

DATA= SAS® data set that contains the table of interest.
ROW= The name of the single variable specifying the rows in the table.
COL= The name of the single variable specifying the columns in the table.

The following parameter is optional:

COUNT= The name of the variable that contains the cell counts. This only needs to be specified if the input data set is already aggregated.

The version of the %RUN_FISHERS macro that you are using is displayed when you specify version (or any string) as the first argument. For example:

```

/*****
%run_fishers (version, data=a, row=r, col=c)
*****/

```

The %RUN_FISHERS macro can only be used for a single table request. Specifying multiple rows and/or multiple columns (corresponding to a multiple table request such as (a b)*(c d) in the TABLE statement of PROC FREQ) is not supported. No error checking is done. The macro assumes that the specified data set and variables exist and that their names are correctly specified. Note that Fisher's exact test can become very computationally intensive as the sample size and the dimension of the table become large. This can result in extremely long run times and very large memory requirements.

THE EXAMPLE:

This example checks to see if at least 80% of the cells in a 2x3 table have an expected value greater than 5 and then runs Fisher's exact test when it fails this test.

```

/*****
data test;
  input r c ct;
  datalines;
  1 1 12
  1 2 5
  1 3 6
  2 1 13
  2 2 2
  2 3 4
  ;
run;

/* Define the RUN_FISHERS macro */
%inc "<location of your file containing the RUN_FISHERS macro>";

%run_fishers(data=test,
             row=r,
             col=c,
             count=ct)
*****/

```

THE RESULTS

The %RUN_FISHERS macro produces the following results since only 50% of the cells have an expected value greater than 5.

This MACRO automatically performs Fisher's exact test whenever the chi-square test results in a WARNING message regarding the validity of the test in an RxC table because of a large number of cells with expected count less than 5.

THE STATEMENTS of MACRO

```

/*****
%macro run_fishers (version, data=, row=, col=, count=);

%let _version=1.0;

%if &version ne %then %put RUN_FISHERS macro Version &_version;

%let opts = %sysfunc(getoption(notes))
            _last_=%sysfunc(getoption(_last_));

%if &version ne debug %then %str(options nonotes);

/* Check for newer version */
%if %sysevalf(&sysver >= 8.2) %then %do;
  %let _notfound=0;

filename ver url 'http://ftp.sas.com/techsup/download/stat/versions.dat' termstr=crlf;
*****/

```

```

data _null_;
  infile ver end=_eof;
  input name:$15. ver;
  if upcase(name)="&sysmacroname" then do;
    call symput("_newver",ver); stop;
  end;

  if _eof then call symput("_notfound",1);
run;
%if &syserr ne 0 or &_notfound=1 %then
  %put &sysmacroname: Unable to check for newer version;
%else %if %sysevalf(&_newver > &_version) %then %do;
  %put &sysmacroname: A newer version of the &sysmacroname macro is available.;
  %put %str(          ) You can get the newer version at this location;;
  %put %str(          ) http://support.sas.com/ctx/samples/index.jsp;
%end;
%end;

proc freq data=&data noprint;
  %if &count ne %then %str(weight &count / zeros);)
  tables &row*&col / sparse outexpect out=_out1;
run;

proc means data=_out1 noprint;
  var count;
  output out=_out2;
run;

data _null_;
  set _out1;
  if expected<=5 then warn+1;
  if _n_=1 then set _out2;
  pct_lt5=warn/_freq;
  if _freq=_n;
  warning=(pct_lt5>=.2);
  call symput('warning',warning);
run;

options &opts;

proc freq data=_out1;
  weight count / zeros;
  tables &row*&col / chisq;
  %if &warning=1 %then %do;
  exact fisher;
  %end;
run;

%mend;

/*****/

```

CONCLUSION

This is a simple way to automatically performs Fisher's exact test whenever a warning message for the Chi-Square test is issued.

The advantages are:

- (1) The new macro call can be created to automatically run Fisher's exact test when the chi-square test might not be valid.
- (2) This macro %RUN_FISHERS should be located in a central place such as project macro area and each program can call the macro from there.

ACKNOWLEDGEMENTS

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

CONTACT INFORMATION

Wei Xu

Boston Scientific

100 Boston Scientific Way

Marlborough, MA 01752-1234

(508) 683-4264

wei.xu@bsci.com