EXACTing a Price: Compute Fisher's Exact Test P-values Only When Needed
Robert Abelson, Human Genome Sciences, Rockville, Maryland

ABSTRACT
Many times when p-values are computed for a categorical analysis, it is not known in advance whether an exact or an asymptotic p-value is needed, so both are computed using the EXACT option in PROC FREQ. Sometimes SAS® issues a warning that “Computing exact p-values may require much time and memory.” It would be helpful to compute exact p-values only when they are needed. The use of Fisher’s Exact Test is recommended when expected cell counts of less than 5 comprise 25% or more of a table. Some statisticians are more conservative, favoring a lower cutoff percentage. Also, while many statisticians use the Pearson chi-square p-values when exact p-values are not needed, some prefer the likelihood ratio chi-square. A macro is presented which gets the expected cell counts, determines if an exact p-value is needed, computes exact p-values only when the number of cells with expected counts less than 5 exceeds the cutoff percentage, and computes chi-square (either Pearson chi-square or likelihood ratio test) p-values for the remaining tables.

INTRODUCTION
In the past, we would calculate both the Fisher’s Exact Test and the chi-square p-values, then check the expected cell counts to see which one to choose to include in a summary table. This would sometimes result in SAS issuing the warning “Computing exact p-values may require much time and memory” and the program would have to be aborted. Here is an example of code that was used which would get the correct result, when it ran to completion:

```sas
proc freq data=p_cnt;
   by maj order name ;
   tables trt*yn/chisq fisher sparse expected outexpect out=out1; * output expected cell count to data out1;
   output pchi fisher out=pvalout;
   weight cnt;
run;

*** Number of cells;
proc means data=out1 noprint;
   var count;
   by maj order name;
   output out=out2 n=n;
run;

*** Datastep warn to flag low cell count in frequency tables;
data warn;
   set out1;
   by maj order name;
   if first.name then warn=0; * Initialize warning count;
   if expected<5 then warn+1;
   if first.name then set out2;
   pct_lt5 = warn/_freq_;  * Percentage of frequency less than 5;
   warning=(pct_lt5>=.25); * If >=25% of the cells have expected counts less than 5, warning is set to 1;
   if last.name;
      keep maj order name warning;
run;

data pvalout;
   merge pvalout(keep=maj name order p_pchi xp2_fish) warn;
   by maj order name;
   if warning = 1 then _pvalue=xp2_fish;
   else _pvalue=p_pchi;
   attrib _pvalue format=pvalfmt. label='P-value';
   drop warning p_pchi xp2_fish;
run;
```
This computes the expected cell count, chi-square, and Fisher's Exact Test, then chooses the chi-square or Fisher's Exact Test p-value as appropriate. This is not a problem with 2x2 contingency tables, but does become a problem when more rows or columns are added.

**A BETTER APPROACH**

It would be helpful to compute exact p-values only when they are needed. The use of Fisher's Exact Test is recommended when expected cell counts of less than 5 comprise 25% or more of a table. Some statisticians are more conservative, favoring a lower cutoff percentage. Also, while many statisticians use the Pearson chi-square p-values when exact p-values are not needed, some prefer the likelihood ratio chi-square.

This macro first computes the expected cell counts, then proceeds to compute asymptotic p-values (either Pearson chi-square or likelihood ratio chi-square), or Fisher's Exact Test p-values, if needed. If there are BY variables, each level of BY values is checked to see whether or not the exact test is needed.

```
%macro pvalcat(ds=, /* input dataset */
    rowvar=, /* row variable */
    colvar=, /* column variable */
    wtvar=, /* weight variable */
    frmat=, /* format statement */
    byvars=, /* BY-variables */
    chisqpv=PEARSON, /* Type of chi-square test (PEARSON or LRCHI) */
    pval_cat=, /* Output dataset containing the p-values */
    exppct=.25 /* Proportion of expected cell count less than 5 that triggers FET */
);
/* Is dataset non-empty? If no, do not attempt to calculate p-values. */
data _null_;  
  if 0 then set &ds nobs=nobs;
  call symput('NOBS', put(nobs, best.));
  stop;
run;
%if &nobs>0 %then %do;
/* if there are BY variables, we will need a comma-separated string of them */
  %if &byvars ne %then %do;
    data scratch;
    length byvars byvars_cs $ 200;
    byvars = left(trim(compbl(symget('byvars'))));
    nw = (compress(byvars) ne ' ') * 
      (length(left(trim(compbl(byvars))))- 
        length(left(trim(compress(byvars)))))+1);
    put nw=;
    if nw>1 then byvars_cs = tranwrd(trim(byvars), ' ', ' ,');
    else byvars_cs = byvars;
    call symput('byvars_cs', byvars_cs);
    run;
    proc sort data=&ds;
      by &byvars;
    run;
  %end;
ods listing close;
ods output CrossTabFreqs=ctfreqs; /* cell counts and expected counts */
/* Get number of cells and expected cell counts to determine which test to use */
proc freq data=&ds;
  %if &wtvar ne %then %do;
    weight &wtvar;
  %end;
  %if &byvars ne %then %do;
    by &byvars;
  %end;
```
%if format ne %then %do;
  &format;
%end;
tables &rowvar*&colvar / sparse expected;
run;

proc sql;
/* Get total number of cells */
create table ncells as
select
  %if &byvars ne %then %do;
    &byvars_cs,
  %end;
count(*) as numcells
from ctfreqs
where index(_type_, '0')=0 and expected ne .
%if &byvars ne %then %do;
  group by &byvars_cs
  order by &byvars_cs
%end;
;
/* Get number of cells with expected cell count less than 5 */
create table lowexpect as
select
  %if &byvars ne %then %do;
    &byvars_cs,
  %end;
count(*) as numlow
from ctfreqs
where index(_type_, '0')=0 and expected ne . and expected<5
%if &byvars ne %then %do;
  group by &byvars_cs
  order by &byvars_cs
%end;
;
quit;

/* Decide whether Fisher Exact Test needs to be done */
data fishflag;
merge ncells lowexpect;
/* OK to merge without BY if there are no BY-variables */
%if &byvars ne %then %do;
  by &byvars;
%end;
if numlow ne . and numlow/numcells>=&exppct then usefish=1;
else usefish=0;
run;

/* See if there are any levels that need to use Fisher's Exact Test */
proc sql noprint;
select count(*) into :fishcnt
from fishflag
where usefish=1;
quit;
%if &fishcnt>0 %then %do;
  %if &byvars ne %then %do;
    data &ds.2;
    merge &ds fishflag(in=inf);
    by &byvars;
    if inf;
    run;
  %end;
%else %do;
  data fishflag;
  set fishflag;
  dummy=1;
  run;
%end;
data &ds;
   set &ds;
   dummy=1;
run;

data &ds.2;
   merge &ds fishflag(in=inf);
   by dummy;
   if inf;
run;
%end;
ods output FishersExact=fisherstest; /* Fisher Exact Test */
proc freq data=&ds.2;
   %if &wtvar ne %then %do;
   weight &wtvar;
   %end;
   %if &byvars ne %then %do;
   by &byvars;
   %end;
   %if frmat ne %then %do;
   &frmat;
   %end;
   where usefish=1;
   tables &rowvar*&colvar / sparse fisher;
run;
%end;
ods output ChiSq=chisquare; /* chi-square statistics */
proc freq data=&ds;
   %if &wtvar ne %then %do;
   weight &wtvar;
   %end;
   %if &byvars ne %then %do;
   by &byvars;
   %end;
   %if frmat ne %then %do;
   &frmat;
   %end;
   tables &rowvar*&colvar / sparse chisq;
run;
/* Skip the rest of the calculations if chisquare does not exist. */
%if %sysfunc(exist(work.chisquare)) %then %do;
   /* Choose which chi-square p-value to use (Pearson or likelihood ratio) based on &CHISQPV */
   proc sql;
   create table pv_chisq as
   select
      %if &byvars ne %then %do;
      &byvars_cs ,
      %end;
   prob as p_chisq
   from chisquare
   %if &chisqpv=LRCHI %then %do;
   where statistic='Likelihood Ratio Chi-Square'
   %end;
   %else %do;
   where statistic='Chi-Square'
   %end;
;
%if %sysfunc(exist(work.fisherstest)) %then %do;
   create table pv_fisher as
   select
      %if &byvars ne %then %do;
      &byvars_cs ,
      %end;

EXACTing a Price: Compute Fisher's Exact Test P-values Only When Needed, continued

&byvars_cs ,
%end;
nvalue1 as p_fisher
from fisherstest
where name1='XP2_FISH';
%end;
quit;

/*
This next step is needed if there are BY-variables because some (but not all)
levels of a BY-variable may need the Fisher's Exact Test.
*/
data &pval_cat;
%if %sysfunc(exist(work.pv_fisher)) %then %do;
merge pv_chisq(in=in_chi) pv_fisher(in=in_fish);
%end;
%else %do;
set pv_chisq;
%end;
%if &byvars ne %then %do;
by &byvars;
%end;
%if %sysfunc(exist(work.pv_fisher)) %then %do;
if in_fish then pvalue = p_fisher;
else pvalue = p_chisq;
%end;
%else %do;
pvalue = p_chisq;
%end;
keep %if &byvars ne %then %do; &byvars %end; pvalue;
run;
%end;
%else %do;
%if &byvars ne %then %do;
proc sql;
create table &pval_cat as
select distinct &byvars_cs, . as pvalue
from &ds;
quit;
%end;
%else %do;
data &pval_cat;
pvalue=.;
run;
%end;
%end;
/* Clean up. */
proc datasets nolist library=work;
delete
%if &byvars ne %then %do; scratch %end;
%if %sysfunc(exist(CTFREQS)) %then %do; ctfreqs %end;
%if %sysfunc(exist(NCELLLS)) %then %do; ncells %end;
%if %sysfunc(exist(LOWEXPECT)) %then %do; lowexpect %end;
%if %sysfunc(exist(FISHFLAG)) %then %do; fishflag %end;
%if %sysfunc(exist(USEFISH)) %then %do; usefish %end;
%if %sysfunc(exist(USECHI)) %then %do; usechi %end;
%if %sysfunc(exist(FISHERSTEST)) %then %do; fisherstest %end;
%if %sysfunc(exist(CHISQUARE)) %then %do; chisquare %end;
%if %sysfunc(exist( %upcase(&ds.2))) %then %do; &ds.2 %end;
%if %sysfunc(exist(PV_CHAN)) %then %do; pv_chisq %end;
%if %sysfunc(exist(PV_FISHER)) %then %do; pv_fisher %end;
run;
quit;
ods listing;
%end;
%else %do;
    data _null_;  
       put "Dataset &ds has 0 observations, therefore, p-values cannot be calculated.";  
         run;  
%end;  
%symdel ds rowvar colvar wtvar fmtvars chispv pval_cat exppct nobs fishcnt byvars_cs / nowarn;  
%end pvalcat;

In addition to avoiding unneeded calculations, this has the advantage of being a macro and thus reusable, and also takes advantage of the SAS Output Delivery System. There have been far fewer occurrences of the warning about excessive computation time for exact p-values.

CONCLUSION

To avoid situations where Fisher’s Exact Test is calculated unnecessarily, get the expected cell counts first, and then proceed with asymptotic or exact p-value calculations, as needed.

CONTACT INFORMATION

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

    Name: Robert Abelson  
    Enterprise: Human Genome Sciences  
    Address: 14200 Shady Grove Road  
    City, State ZIP: Rockville, MD 20850  
    Work Phone: 240 314 4400 x1374  
    Fax: 301 279 8799  
    E-mail: bob_abelson@hgsi.com  
    Web: www.hgsi.com

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.

Other brand and product names are trademarks of their respective companies.