A SAS ® PROGRAM FOR DETERMINING DISTRIBUTION-FREE CONFIDENCE INTERVALS FOR MEDIANS

Willard Carl Losinger

Bureau of the Census

I. ABSTRACT

A SAS ® program for determining confidence intervals for population medians from a simple random sample (without replacement) from the population is presented. The population size is assumed to be known. The background of the problem is discussed. An application of the PROBHYPR function is revealed.

II. BACKGROUND

William R. Thompson [1] defines a finite population, U_N, of real numbers $|x^{(i)}|$, $x^{(i)} < x^{(j)}$ for $i < j$, $i = 1, \ldots, N$. N is assumed to be known, and a random sample of n values is drawn from U_N without replacement. The sample values are denoted $\{x_k\}$, $k = 1, \ldots, n$; where k is the order of ascending magnitude in the sample. Each $x_k = x^{(u(k))}$ for some unknown $u_k = 1, 2, \ldots, N$.

In general, the expected probability that R (the number of values which are less than x_k in U_N) lies between u_k and u_{k+1} can be determined by

$$\overline{P}(u_k \le R < u_{k+1}) = \frac{\binom{R}{k}\binom{N-R}{n-k}}{\binom{N}{n}} \tag{1}$$

Equation (1) is recognized as having a hypergeometric distribution.

Letting M represent the unknown population median and m = (N - a) / 2 where a = 0 or 1 so that N - a is even (in other words, m represents the number of population values which are less than the population median),

$$\overline{P}(x_k \le M < x_{k+1}) = \overline{P}(u_k \le m < u_{k+1}) = \frac{\binom{m}{k}\binom{N-m}{n-k}}{\binom{N}{n}} \tag{2}$$

gives the expected probability that the population median lies between two given sequential values in the sample.

$$\overline{P}(x_i \le M < x_j) = \overline{P}(u_i \le m < u_j) = \sum_{k=i}^{j-1} \frac{\binom{m}{k}\binom{N-m}{n-k}}{\binom{N}{n}} \tag{3}$$

where $i < j$, $i = 1, \ldots, n$, gives good results in determining confidence ranges for the median.

Table 1 lists some ninety percent confidence intervals (which were derived from equation (3)) for simple random samples from various population sizes. For example, for a random sample of size 30 from a population of size 180, Table 1 indicates that we can be 90% confident that the population median lies between the eleventh and twentieth sample values when the sample values have been arranged in ascending order of magnitude. This result was examined empirically by Alfredo Navarro [2] using a Monte Carlo resampling technique on some income data and found to hold true.

As N becomes large, the hypergeometric function converges to the binomial probability function. Tables for confidence intervals of the median for sample sizes to 1000 and for large population sizes have been produced by William J. MacKinnon [3].

III. THE PROGRAM

This SAS ® program starts from the middle of a sample distribution (where the population size is known) and cumulates expected probabilities outward (using Equation 2) until either the desired level of confidence or the end of the sample is attained.

The variable MEDSAMP represents the sample rank of the sample median (or the sample rank of the sample value just below the sample median for even sized samples). First, the expected probability that the population median lies between MEDSAMP and the next higher sample value is determined. The PROBHYPR function [4] returns the cumulative probability from a hypergeometric distribution. Subtracting the value returned from the PROBHYPR function at MEDSAMP-1 from the value returned from the PROBHYPR function at MEDSAMP yields the desired result.

Expected probabilities are then cumulated incrementally both above (using K2 and PBAR2) and below (using K1 and PBAR1) the sample median until the desired level of confidence is attained. The following principle is applied:

Multiplying Equation (2) by the factor

$$\left(\frac{m-k}{k+1} \right) \left(\frac{n-k}{N-m-n+k+1} \right)$$

gives the result for $\bar{P}(x_{k+1} \leq M < x_{k+2})$. The need to use the PROBHYPR function again is obviated, thus allowing the program to run more quickly. Similarly, multiplying Equation (2) by the factor

$$\left(\frac{k}{m-k+1} \right) \left(\frac{N-m-n+k}{n-k+1} \right)$$

yields the result for $\bar{P}(x_{k-1} \leq M < x_k)$.

IV. FURTHER APPLICATIONS

A SAS ® user who has a sorted sample data set and knows the population and sample size may output the variables BOTINT and TOPINT from this program, and then use the POINT= option in a SET statement [5] to create a data set which would contain the limits of the desired confidence interval for the population median.

If dealing with a sorted sample data set containing weighted data, one could create a new variable (called CUMWGT) based upon the cumulated weights. The sample values associated with the values for CUMWGT which are closest to the product of the results from this program times the average weight should give a reasonable estimate for the confidence interval for the median.

Slightly modifying this program could make it useful in constructing confidence intervals for any percentile or any ordered value in the population--not just the median. For example, by setting m = FLOOR(NPOP*0.30) and MEDSAMP = FLOOR(NSAMP*0.30), the program would generate confidence intervals for the thirtieth percentile in a population. By setting m = 32 and MEDSAMP = FLOOR(32*NSAMP/NPOP), this program would determine confidence intervals for the thirty-second ordered value in populations greater than size 32.

V. FOOTNOTES

1. Thompson, William R. (1936). On confidence ranges for the median and other expectation distributions for populations of unknown distribution form. Annals of Mathematical Statistics 7, 122-128.

2. Navarro, Alfredo (1991). Reliability of the sample median for income distributions. STSD 1990 DECENNIAL CENSUS MEMORANDUM SERIES, #MM-10. (Internal document, Bureau of the Census).

3. MacKinnon, William J. (1964). Table for both the sign test and distribution-free confidence intervals of the median for sample sizes to 1,000. American Statistical Association Journal 59, 935-956.

4. SAS Institute Inc. SAS ® Language: Reference, Version 6, First Edition. Cary, NC: SAS Institute, Inc., 1990. Pages 580-581.

5. Ibidem, page 485.

TABLE 1

90% Confidence Intervals for Medians

Sample Size	100% Population Count	Expected Rank of Median	Bottom of Median Confidence Interval	Top of Median Confidence Interval	Probability of Confidence Interval
6,517	65,209	3259.0	3,196	3,323	0.903
306	2,887	153.5	140	168	0.909
10	7,948	5.5	3	9	0.935
30	3,073	15.5	11	20	0.903
229	2,461	115.0	104	128	0.901
497	4,522	249.0	232	267	0.904
30	180	15.5	11	20	0.929
1,700	1,733	850.5	846	856	0.904
7,350	73,414	3675.5	3,609	3,743	0.901

```
*********************MEDIANS.SAS**********************************;

options pagesize=62 linesize=70 nonumber nodate ;

data medians ;                   /* THE SAMPLE SIZES ARE ENTERED */
input nsamp npop ; cards ;
6517 65209
306 2887
10 7948
30 3073
229 2461
497 4522
30 180
1700 1733
7350 73414
;
data medians ; set medians ;

clevel = 0.90 ;     /* SET THE DESIRED LEVEL OF CONFIDENCE */

m=floor(npop/2) ;   /* THE NUMBER OF VALUES BELOW THE MEDIAN IN THE
                       POPULATION  */

medsamp=floor((nsamp+1)/2) ; /* AT OR JUST BELOW THE SAMPLE MEDIAN */

pbar=probhypr(npop,m,nsamp,medsamp)-probhypr(npop,m,nsamp,medsamp-1) ;
                /* THE EXPECTED PROBABILITY THAT THE POPULATION MEDIAN
                   IS BETWEEN THE MEDSAMP-TH AND THE (MEDSAMP+1)-TH
                   SAMPLE UNITS) */
if pbar eq . then stop ;
```

```
**********************************************************
THE LOOPS RUN UNTIL EITHER PBAR GETS TO CLEVEL OR THE END
OF THE SAMPLE IS ATTAINED.
********************************************************** ;

k1 = medsamp ; /* K1 AND PBAR1 LOOK AT PROBABILITIES BELOW THE */
pbar1 = pbar ; /* SAMPLE MEDIAN */
k2 = medsamp ; /* K2 AND PBAR2 LOOK AT PROBABILITIES ABOVE THE */
pbar2 = pbar ; /* SAMPLE MEDIAN */
limit = nsamp - 1 ; /* THE MAXIMUM VALUE FOR K2 */

if (pbar lt clevel and (k1 gt 1 or k2 lt limit))
then do until(pbar ge clevel or (k1 le 1 and k2 ge limit)) ;

 if (k2 lt limit) then do ;
  pbar2 = pbar2 * ((m-k2)/(k2+1)) * ((nsamp-k2)/(npop-m-nsamp+k2+1)) ;
  pbar = pbar + pbar2 ;
  if pbar eq . then stop ;
  k2 = k2 + 1 ;
 end ;

 if (pbar lt clevel and k1 gt 1) then do ;
  pbar1 = pbar1 * (k1/(m-k1+1)) * ((npop-m-nsamp+k1)/(nsamp-k1+1)) ;
  pbar = pbar + pbar1 ;
  if pbar eq . then stop ;
  k1 = k1 - 1 ;
 end ;
end ;

botint = k1 ;                   /* BOTTOM OF C.I. */
topint = k2 + 1 ;              /* TOP OF C.I. */
range = topint - botint + 1 ;  /* RANGE OF C.I. */
median = (nsamp+1) / 2 ;       /* MEDIAN OF SAMPLE */
ratio = range / median ;       /* RANGE TO MEDIAN RATIO */

proc print noobs label ;
var nsamp npop median botint topint pbar ;
label nsamp = 'Sample Size'
      npop = '100% Population Count'
      median = 'Expected Rank of Median'
      botint = 'Bottom of Median Confidence Interval'
      topint = 'Top of Median Confidence Interval'
      range = 'Range of Confidence Interval'
      ratio = 'Range to Median Ratio'
      pbar = 'Probability of Confidence Interval' ;
format ratio 4.2 pbar 5.3 nsamp npop botint topint comma10. ;
title1 'TABLE 1' ;
title3 '90% Confidence Intervals for Medians' ;
```