

Paper 230-2008

ALPHA_CI: A SAS® Macro for Computing Confidence Intervals for Coefficient Alpha

Jeffrey D. Kromrey, University of South Florida

Jeanine Romano, The American University of Sharjah

Susan T. Hibbard, University of South Florida

ABSTRACT

Coefficient alpha is a commonly used index of measurement reliability in the social and behavioral sciences. Although point estimates of coefficient alpha are readily computed (and are provided by the PROC CORR procedure in SAS), a variety of methods for constructing interval estimates have been suggested in the literature. This paper presents a SAS macro that calculates eight interval estimates of coefficient alpha. The macro computes coefficient alpha from a matrix of item scores provided as input, and outputs the point estimate and interval estimates in a simple table. The paper provides a demonstration of the SAS/IML code, sample output, and examples of applications in simulation studies.

INTRODUCTION

An important index of measurement quality is the degree to which a test score is consistent. In classical test theory, the reliability coefficient, ρ_{xx} , is defined as the correlation between scores on parallel tests (Crocker & Algina, 1986). According to classical test theory, an examinee's observed score, X , can be expressed as the sum of his/her true score and random error: $X = T + E$.

The reliability coefficient is the proportion of the observed variance in scores that represents true score variance rather than random error:

$$\rho_{xx} = \frac{\sigma_{true}^2}{\sigma_{total}^2}$$

The most common approaches for estimating the reliability of scores include administering the same test twice to the same examinees (test-retest reliability) or administering the test once and estimating score reliability from the intercorrelation of test items (internal consistency reliability) (Crocker & Algina, 1986). Several indices can be used to measure internal consistency, the most common being coefficient alpha, sometimes referred to as Cronbach's alpha (Cronbach, 1951; Hogan, Benjamin & Brezinski, 2000; Peterson, 1994):

$$\hat{\alpha} = \frac{k}{k-1} \left(1 - \frac{\sum_{i=1}^k \sigma_i^2}{\sigma_x^2} \right)$$

Where k is the number items on the test, σ_i^2 is the variance of the i^{th} item, and σ_x^2 is the variance of the total (summed across items) scores (Crocker & Algina, 1986).

ALTERNATIVE METHODS FOR COEFFICIENT ALPHA CONFIDENCE INTERVALS

The sampling distribution of coefficient alpha has been discussed in the literature for many years (Feldt & Brennan, 1989; Felt 1995; Kristoff, 1963; Lord, 1974) and several approaches for confidence interval estimation have been recommended (see, for example, Bonett, 2002; Feldt 1965; Hakstian & Whalen, 1976; Iacobucci &

Duchachek, 2003; Koning & Franses, 2003).

Procedures Based on Transformations of Alpha. Several confidence interval procedures are derived by using a normalizing transformation of the sample alpha. The logic of these methods is that a suitably transformed statistic will be approximately normally distributed, allowing the use of normal theory for the construction of confidence intervals. The endpoints of the resulting confidence interval are subsequently back-transformed to the scale of coefficient alpha. For example, Bonett (2002) suggested the transformation $z = \ln(1 - \hat{\alpha})$. The resulting statistic is reported to be approximately normally distributed with a variance of $2k\{(k-1)(n-2)\}$, where k = the number of items on the instrument. In contrast, Hakstain and Whalen (1976) recommended the transformation $z = \ln(1 - \hat{\alpha})^{1/3}$. This transformed value is approximately normally distributed with a variance of $\left[18k(n-1)(1-\hat{\alpha})^{2/3}\right] / \left[(k-1)(9n-11)^2\right]$.

Other researchers have used the Fisher z transformation that is often used to normalize distributions of correlation coefficients:

$$z = \frac{1}{2} \ln \left(\frac{1 + |\hat{\alpha}|}{1 - |\hat{\alpha}|} \right) \quad \text{or} \quad z = \tanh^{-1} \hat{\alpha}$$

This transformed statistic is approximately normally distributed with a variance of $1/(n-3)$.

Procedures that Avoid Transformations of Alpha. Additional confidence interval procedures have been derived that avoid normalizing transformation of the sample alpha. For example, Feldt (1965) uses an interval inversion approach based upon the F distribution. A similar (but not identical) result was independently derived by Koning and Franses (2003). The latter authors also suggested an asymptotic confidence interval for coefficient alpha, the endpoints of which are given by:

$$\hat{\alpha}_L = \hat{\alpha} - (1 - \hat{\alpha}) \exp \left\{ z_L \sqrt{\frac{2k}{n(k-1)}} \right\} \quad \text{and} \quad \hat{\alpha}_R = \hat{\alpha} - (1 - \hat{\alpha}) \exp \left\{ z_R \sqrt{\frac{2k}{n(k-1)}} \right\}$$

Additionally, Iacobucci and Duchachek (2003) derived confidence intervals based upon the item covariance matrix:

$$\left| \frac{2k^2}{(k-1)^2} \right| j' V j (\text{trace}(V^2) + (\text{trace}(V))^2 - 2(\text{trace}(V)(j' V^2 j)))$$

Finally, Maydeu-Olivares, Coffman, and Hartmann (2007) derived an asymptotically distribution free (ADF) interval estimate for coefficient alpha. Results of their simulations suggest this interval provides superior confidence interval coverage probabilities when item distributions are not normal.

MACRO ALPHA_CI

A SAS/IML macro was designed to compute the sample value of coefficient alpha and confidence intervals for alpha using each of the methods for interval estimation described above. The macro was developed to provide researchers with an easily accessible tool for calculating interval estimates for alpha. Inputs to the macro include the name of the SAS dataset containing the sample of observed item responses and the number of items to be included. As written, the names of the item variables in the SAS dataset must be X1, X2, X3, etc.

```
%macro ALPHA_CI (dsn = _last_, n_items = 100, confidence = .95);
proc iml symsize = 500;
```

```
start Bonett(N1, items, rxx, confidence, lowerCI, upperCI);
```

```
* +-----+
Bonett 2002 method of CI computation
```

```
Inputs to subroutine are number of examinees (N1), number of items (items),
```

sample value of coefficient alpha (rxx), and desired level of confidence (Confidence).

Outputs are the upper and lower limits of the confidence interval (upperCI, lowerCI)

```
+-----+
critZ = -1#probit((1-Confidence)/2);
transBon_z = log(1-abs(rxx));
if rxx <0 then transBon_z = transBon_z * -1;
var_Z = (2#items)/((items - 1) # (n1 - 2));
SE_Bon_Z = sqrt(var_z);
lowerCI = 1 - exp(transBon_z + critz#SE_Bon_z);
upperCI = 1 - exp(transBon_z - critz#SE_Bon_z);
finish;
```

Start Feldt(n1,items,rxx,Confidence,lowerCI,upperCI);

```
* +-----+
Fel dt (1965) method of CI computation
```

Inputs to subroutine are number of examinees (N), number of items (items), sample value of coefficient alpha (alpha), and desired level of confidence (Confidence).

Outputs are the upper and lower limits of the confidence interval (upperCI, lowerCI)

```
+-----+
lowerF = (1-Confidence)/2;
upperF = Confidence + lowerF;
Fb = FINV(upperF,N1-1,(N1-1)#(items-1),0);
Fa = FINV(lowerF,N1-1,(N1-1)#(items-1),0);
lowerCI = 1 - (Fb)#(1-rxx);
upperCI = 1 - (Fa)#(1-rxx);
finish;
```

start Fisher(N1,items,rxx,confidence,lowerCI,upperCI);

```
* +-----+
Fisher z transformation method of CI computation
```

Inputs to subroutine are number of examinees (N), number of items (items), sample value of coefficient alpha (alpha), and desired level of confidence (Confidence).

Outputs are the upper and lower limits of the confidence interval (upperCI, lowerCI)

```
+-----+
critZ = -1#probit((1-Confidence)/2);
transfish_z = 0.5#log((1+abs(rxx))/(1-abs(rxx)));
if rxx <0 then transfish_z = transfish_z * -1;
SE_fish_z = sqrt((n1-3)##-1);
upperCI = (exp(2#(transfish_z + critZ#SE_fish_z)) - 1) /
           (exp(2#(transfish_z + critZ#SE_fish_z)) + 1);
lowerCI = (exp(2#(transfish_z - critZ#SE_fish_z)) - 1) /
           (exp(2#(transfish_z - critZ#SE_fish_z)) + 1);
finish;
```

start HW(n1,items,rxx,Confidence,lowerCI,upperCI);

```
* +-----+
Hakstain & Whalen (1976) method of CI computation
```

Inputs to subroutine are number of examinees (n1), number of items (items), sample value of coefficient alpha (alpha), and desired level of confidence (Confidence).

Outputs are the upper and lower limits of the confidence interval (upperCI, lowerCI)

```
+-----+
critZ = -1#probit((1-Confidence)/2);
transHaWh_z = (1-rxx)**(1/3);
var_z = ((18#items)#(n1 - 1)#((1 - rxx)**(2/3)))/((items - 1)#((9#n1 - 11)**(2)));
SE_HaWh_z = sqrt(var_z);
c_star = ((9#n1 - 11)#(items - 1))/(9#(n1 - 1)#(items - 1) - 2);
lowerCI = 1 - c_star###3#(((1-rxx)**(1/3) + CritZ#SE_HaWh_z)###3);
upperCI = 1 - c_star###3#(((1-rxx)**(1/3) - CritZ#SE_HaWh_z)###3);
finish;
```

```
start ID(N1,items,rxx,out3pl,confidence,lowerCI,upperCI);
```

```
* +-----+
Iacobucci and Duchachek (2003) transformation method of CI computation
Inputs are the nunnber of examinees(N1), the number of items (items) the
outputs are the upper and lower limits of the confidence interval (upperCI, lowerCI)
+-----+;
```

```
CritZ = -1#probit((1-Confidence)/2);
bigone = j(N1, 1);
means = ((bigone`)*out3pl)/N1;
SCORE3PLd = out3pl-(bigone*means);
itemcov = (1/(N1-1))*((SCORE3PLd`)*SCORE3PLd);
one = j(items,1);
jtphij = (one`)*itemcov*one;
trphisq = trace(itemcov*itemcov);
trsqphi = (trace(itemcov))**2;
jtphisqj = (one`)*(itemcov*itemcov)*one;
omega = jtphij*(trphisq+trsqphi);
omega = omega-(2*(trace(itemcov))*jtphisqj);
omega = (2/(jtphij**3))*omega;
s2 = (items**2)/((items-1)**2);
s2 = s2*omega;
se = sqrt(s2/N1);
LowerCI = rxx-(CritZ*se);
UpperCI = rxx+(CritZ*se);
```

```
finish;
```

```
start KF1(n1,items,rxx,Confidence,lowerCI,upperCI);
```

```
* +-----+
Koning and Frances (2003) 'Exact' method of CI computation

Inputs to subroutine are number of examinees (n1), number of items (items),
sample value of coefficient alpha (rxx), and desired level of confidence
(Confidence).
```

Outputs are the upper and lower limits of the confidence interval (upperCI, lowerCI)

```
+-----+
lowerF = (1-Confidence)/2;
```

```

    upperF = Confidence + lowerF;
    Fb = FINV(upperF,n1#(items-1),n1,0);
    Fa = FINV(lowerF,n1#(items-1),n1,0);
    upperCI = 1 - (1-rxx)/Fb;
    lowerCI = 1 - (1-rxx)/Fa;
finish;

start KF2(n1,items,rxx,Confidence,lowerCI,upperCI);
* +-----+
    Koning and Frances (2003) 'Asymptotic' method of CI computation

    Inputs to subroutine are number of examinees (n1), number of items (items),
        sample value of coefficient alpha (rxx), and desired level of confidence
        (Confidence).

    Outputs are the upper and lower limits of the confidence interval (upperCI, lowerCI)
+-----+;
    CritZ = -1#probit((1-Confidence)/2);
    upperCI = 1 - (1-rxx)#exp(-1#CritZ#SQRT((2#items)/(n1#(items - 1))));
    lowerCI = 1 - (1-rxx)#exp(1#CritZ#SQRT((2#items)/(n1#(items - 1))));
finish;

* +-----+
    Maydeu-Olivares, Coffman, and Hartmann (2007) Asymptotically distribution free method
    Note: SAS code for this subroutine was provided by Maydeu-Olivares, Coffman, & Hartmann

    Inputs to subroutine are the item score matrix (data) and desired level of
    confidence (prob).

    Outputs are the upper and lower limits of the confidence interval (Parm[6],
    Parm[7])
+-----+;
start scalpha(data,prob);
    parm = J(1,7,.); /* create return vector: row */
    nsub = NROW(data); /* number of subjects */
    nvar = NCOL(data); /* number of variables */
    nv2 = nvar*(nvar+1)/2;
    rsub = 1. / nsub; rs1 = 1. / (nsub - 1.);
    vrat = nvar / (nvar-1.);
    mean = data[+,] / nsub; /* row vector */
    quant = -1#probit((1-prob)/2);
    cov = rs1 * (data` * data - nsub * mean` * mean);
    vvar = vecdiag(cov); /* variances in diag of cov */
    summat = cov[+,+]; /* sum entire cov matrix */
    sumvars = vvar[+]; /* sum diagonal of cov matrix */
    sumcovs = .5*(summat - sumvars); /* sum off diag of cov matrix */
    wcv = cov * cov;
    wv = vecdiag(wcv);
    sumvar2 = wv[+];
    summat2 = wcv[+,+];
    alpha = vrat * (1. - sumvars/summat);
    parm[1] = alpha; /* alpha */
    tmp = 2. * vrat*vrat / (summat*summat*summat);
    t1 = summat * sumvar2;

```

```

t2 = summat * sumvars*sumvars;
t3 = 2.*sumvars * summat2;
nase = tmp*(t1 + t2 - t3);
nase = sqrt(rsub*nase);
parm[2] = nase; /* NT standard error */
parm[3] = alpha - quant * nase; /* NT CI lower */
parm[4] = alpha + quant * nase; /* NT CI upper */
dwrtvar = -2. * vrat * sumcovs / (summat*summat);
dwrtcov = vrat * sumvars / (summat*summat);
jac = J(nvar,nvar,dwrtcov);
do j= 1 to nvar;
  jac[j,j] = dwrtvar;
end;
trac = 0.;
do isub=1 to nsub;
  v = data[isub,] - mean; /* row vector */
  wcv = jac # (v` * v - cov);
  tmp = wcv[+,+];
  trac = trac + tmp * tmp;
end;
nnase = sqrt(rsub*rs1*trac);
parm[5] = nnase; /* ADF standard error */
parm[6] = alpha - quant * nnase; /* ADF CI lower */
parm[7] = alpha + quant * nnase; /* ADF CI upper */
return(parm);
finish scalpha;

* +-----+
                                Main program
+-----+;

* +-----+
  Read item scores into IML
+-----+;
  use &dsn;
  %do i = 1 %to &n_items;
    read all var{x&i} into temp_x;
    if &i = 1 then do;
      Item_Scores = temp_x;
    End;
    If &i > 1 then do;
      Item_Scores = Item_Scores || temp_x;
    End;
  %end;
  confidence = &confidence;

* +-----+
  Computation of Cronbach Alpha
+-----+;
  N_items = ncol(Item_Scores);
  N_obs = nrow(Item_Scores);
  mu1 = J(N_items,1,0);* ;
  var = J(1,N_items,0);* ;

  do k = 1 to N_items;

```

```

        do i=1 to N_obs;
            mu1[k,1] = mu1[k,1] + Item_Scores[i,k];* ;
        end;
        var[1,k]=(mu1[k,1]/N_obs)*(1 - mu1[k,1]/N_obs);
    end;
    sumvar=0;
    do k = 1 to N_items;
        sumvar = sumvar + var[1,k];* ;
    end;
    rowsum = J(N_obs,1,0);
    do p = 1 to N_obs;
        do k = 1 to N_items;
            rowsum[p,1]=rowsum[p,1] + Item_Scores[p,k];
        end;
    end;
    end;
    sumscore = 0;
    sumscore2 = 0;
    do p = 1 to N_obs;
        sumscore = sumscore + rowsum[p,1];
        sumscore2= sumscore2 + rowsum[p,1]##2;
    end;
    vartotal= (sumscore2-(sumscore##2/N_obs))/(N_obs);
    * +-----+
      Be sure we have some score variance before going any further
    * +-----+;
    if vartotal <=0 then do;
        print 'Total Score Variance =' vartotal;
        print 'Check the Data!';
    end;
    if vartotal > 0 then do;
        rxx = (N_items/(N_items - 1))*((vartotal- sumvar)/vartotal);
        if rxx < 0.00001 then rxx = .00001;
        if rxx > .99 then rxx = .99;

* +-----+
  Call subroutines for confidence intervals
+-----+;
        run Bonett(N_obs,N_items,rxx,Confidence,lowerBonett,upperBonett);
        run Feldt(N_obs,N_items,rxx,Confidence,lowerFeldt,upperFeldt);
        run Fisher(N_obs,N_items,rxx,Confidence,lowerFisher,upperFisher);
        run HW(N_obs,N_items,rxx,Confidence,lowerHw,upperHW);
        run ID(N_obs,N_items,rxx,Item_Scores,confidence,lowerID,upperID);
        run KF1(N_obs,N_items,rxx,Confidence,lowerKF1,upperKF1);
        run KF2(N_obs,N_items,rxx,Confidence,lowerKF2,upperKF2);
        parm = scalpha(Item_Scores,confidence);
            lowerADF = parm[1,6];
            upperADF = parm[1,7];
    end; * end the 'if vartotal > 0' conditional;
    rxx = round(rxx,.001);

* +-----+
  Printed macro output
+-----+;

```

```

file print;
  put @1 'Confidence Intervals for Coefficient Alpha' /
      @1 '-----' /
      @1 'Level of Confidence:' @30 Confidence /
      @1 'Number of Observations:' @30 N_obs /
      @1 'Number of Items:' @30 N_items /
      @1 'Sample Value of Alpha:' @30 rxx //
      @1 'Method' @40 'Lower' @50 'Upper' /
      @1 '-----' @40 '-----' @50 '-----' /
      @1 'Bonett' @40 lowerBonett 5.3 @50 upperBonett 5.3 /
      @1 'Feldt' @40 lowerFeldt 5.3 @50 upperFeldt 5.3 /
      @1 'Fisher' @40 lowerFisher 5.3 @50 upperFisher 5.3 /
      @1 'Hakstain & Whalen' @40 lowerHW 5.3 @50 upperHW 5.3 /
      @1 'Iacobucci & Duchachek' @40 lowerID 5.3 @50 upperID 5.3 /
      @1 'Koning & Frances Exact' @40 lowerKF1 5.3 @50 upperKF1 5.3 /
      @1 'Koning & Frances Asymptotic' @40 lowerKF2 5.3 @50 upperKF2 5.3 /
      @1 'Asymptotic Distribution Free' @40 lowerADF 5.3 @50 upperADF 5.3 /
      @1 '-----';

quit;
%mend;

```

INVOKING THE MACRO

The easiest way in which the macro ALPHA_CI may be used is to simply create a SAS dataset that inputs the sample item responses. The macro is then called, using as arguments the name of the dataset and the number of items. For example, the following code reads responses to five dichotomous items. The data are read into a SAS dataset called ONE and are referenced by the variable names X1 – X5. Note that the names of the item variables are sequentially numbered with the root of X (the naming convention required in the macro). The call to the macro ALPHA_CI requests the computation of coefficient alpha and the confidence intervals.

```

data one;
  input @1 (X1 - X5)(1.);
cards;
10000
11000
11100
11110
11111
00000
10000
11000
11100
11110
11111
00000
;

%ALPHA_CI (dsn = one,n_items = 5)

run;

```


Note that the specification of a confidence level is not needed if the default level (95% confidence interval) is desired. To request a different level of confidence, the CONFIDENCE = argument is included in the call to the macro. For example, to request 99% and 90% confidence intervals, the following two calls to the macro are used.

```
%ALPHA_CI (dsn = one,n_items = 5,confidence = .99)
```

```
%ALPHA_CI (dsn = one,n_items = 5,confidence = .90)
```

The macro ALPHA_CI does not include methods for handling missing item responses. Therefore, missing data treatments should be applied before invoking the macro (e.g., the deletion of incomplete cases or items, or the use of imputation such as that provided by PROC MI). For comprehensive overviews of missing data treatments, please see Schafer and Graham (2002) and Sinharay, Stern, and Russell (2001).

OUTPUT FROM MACRO ALPHA_CI

The output from the macro is a simple table that presents descriptive statistics about the sample and the endpoints of the confidence intervals computed using the eight methods. Table 1 presents the macro output from the sample data presented above. In addition to the level of confidence for the intervals (0.95), the size of the data matrix (number of observations and number of items) and the sample value of coefficient alpha are provided. For these data, the value of coefficient alpha in the sample was 0.833 based upon five items and 12 observations. The endpoints of the 95% confidence intervals present a variety of interval widths from that of the Fisher interval [0.497, 0.952] to that of the Asymptotic Distribution Free interval [0.737, 0.930].

Table 1

Sample Output from Macro Alpha_CI.

Confidence Intervals for Coefficient Alpha

```
-----
Level of Confidence:          0.95
Number of Observations:      12
Number of Items:             5
Sample Value of Alpha:       0.833
```

Method	Lower	Upper
Bonett	0.556	0.937
Feldt	0.616	0.945
Fisher	0.497	0.952
Hakstain & Whalen	0.636	0.949
Iacobucci & Duchachek	0.691	0.976
Koning & Frances Exact	0.629	0.942
Koning & Frances Asymptotic	0.592	0.932
Asymptotic Distribution Free	0.737	0.930

COMPARISON OF CONFIDENCE INTERVAL ACCURACY AND PRECISION

The accuracy and precision of the confidence intervals produced by these methods were investigated in a Monte Carlo study. The study simulated samples of responses to individual dichotomous items under known and controlled population conditions, calculated coefficient alpha in each sample and estimated each of the confidence intervals. The factors controlled in the simulation study included the population value of alpha, the

number of respondents in the sample, and the number of items in the composite. Ten thousand samples were simulated for each condition examined in the Monte Carlo study. Complete details are provided by Romano, Kromrey, and Hibbard (2008), but partial results are presented here.

Table 2 presents the estimated mean statistical bias in the sample value of coefficient alpha by sample size and population reliability. Some negative bias in alpha was evident in small samples ($n = 10$) and the bias was most severe at the lowest level of population reliability. As sample size increased, however, the bias became negligible regardless of the population value of coefficient alpha.

Table 2

Mean Statistical Bias in Coefficient Alpha by Sample Size and Population Reliability.

Sample Size	Population Reliability			Mean
	0.50	0.70	0.90	
10	-0.08	-0.07	-0.02	-0.06
50	-0.02	-0.01	0.00	-0.01
100	-0.01	-0.01	0.00	0.00
1000	0.00	0.00	0.00	0.00
Mean	-0.03	-0.02	-0.01	-0.02

The differences in confidence interval widths among the eight methods were negligible across the conditions examined. However, the interval widths were dependent on both the number of observations in the sample and the population value of coefficient alpha (Table 3). As expected, the confidence intervals became narrower when they were estimated from larger samples (ranging from an average width of 0.77 for samples of size 10 to an average width of 0.06 for samples of size 1000). Narrower confidence intervals were also obtained with larger values of the population reliability. With low levels of reliability (population value of 0.50), the average confidence interval width was 0.52. When the population reliability was 0.90, the average interval width was only 0.12.

Table 3

Mean 95% Confidence Interval Width by Sample Size and Population Reliability.

Sample Size	Population Reliability			Mean
	0.50	0.70	0.90	
10	1.21	0.80	0.28	0.77
50	0.45	0.29	0.10	0.28
100	0.31	0.20	0.07	0.19
1000	0.10	0.06	0.02	0.06
Mean	0.52	0.34	0.12	0.32

Some differences were evident among the eight methods in the 95% confidence interval coverage probabilities (Table 4). With small samples ($n = 10$), the Asymptotic Distribution Free method showed substantially liberal coverage with only 86% of the sample confidence bands containing the population value of coefficient alpha. Note, however, that the coverage probability converged on the nominal 95% level as the sample size increased. In contrast, the Fisher method of confidence interval estimation was slightly conservative, with average coverage probabilities of .97 - .98 across the sample sizes examined. The Bonett, Feldt, and Hakstain and Whalen methods

of interval estimation showed slightly liberal confidence interval coverage at the smallest sample size examined, but all three methods reached 95% coverage when sample size was at least 50. The Iacobucci and Duchachek method was slightly liberal (93%) with samples of size 10, but coverage reached 95% with larger samples. Finally, the two methods proposed by Koning and Frances (the asymptotic method and the exact method) provided mean confidence interval coverage of 95% across all of the sample sizes examined.

Table 4
Mean 95% Confidence Interval Coverage by Estimation Method and Sample Size.

Estimation Method	Sample Size				Mean
	10	50	100	1000	
Asymptotic Distribution Free	0.86	0.93	0.94	0.95	0.92
Bonett	0.97	0.95	0.95	0.95	0.96
Feldt	0.96	0.95	0.95	0.95	0.95
Fisher	0.98	0.97	0.97	0.97	0.97
Hakstain & Whalen	0.96	0.95	0.95	0.95	0.95
Iacobucci & Duchachek	0.93	0.95	0.95	0.95	0.94
Koning & Frances Asymptotic	0.95	0.95	0.95	0.95	0.95
Koning & Frances Exact	0.95	0.95	0.95	0.95	0.95
Mean	0.94	0.95	0.95	0.95	0.95

CONCLUSIONS

The macro ALPHA_CI is provided to facilitate the use of confidence intervals for coefficient alpha. Specifically, this macro computes confidence intervals using eight methods that have been suggested in the psychometric literature. These intervals are useful for the interpretation of sample estimates of reliability by providing indices of the amount of sampling error expected in the obtained value of alpha. The results of a simulation study that investigated these intervals suggest that some methods provide more accurate confidence intervals, especially with small samples. However, the precision of the confidence intervals (the width of the intervals) was quite consistent across the eight methods. The use of ALPHA_CI is very easy: simply read the item responses or item scores using a regular SAS data step, then call the macro by specifying the data set name, the number of items, and the desired level of confidence for the intervals.

REFERENCES

- Bonett, D. G. (2002). Sample size requirements for testing and estimating coefficient alpha. *Journal of Educational and Behavioral Statistics*, 27, 335 – 340.
- Crocker, L. & Angina, J. (1986). *Introduction to classical and modern test theory*. Toronto: Holt Rinehart & Winston.
- Cronbach, L. J. (1951). Coefficient alpha and the internal structure of tests. *Psychometrika*, 16, 297-334.
- Feldt, L. S. (1965). The approximate sampling distribution of Kuder–Richardson reliability coefficient twenty. *Psychometrika*, 30, 357–370.
- Feldt, L. S., & Brennan, R. L. (1989). Reliability. In R. L. Linn (Ed.), *Educational measurement* (3rd ed., pp. 105–146). New York: American Council on Education and Macmillan.

- Hakstian, A. R., & Whalen, T. E. (1976). A k -sample significance test for independent alpha coefficients. *Psychometrika*, 41, 219–231.
- Hogan, T. P., Benjamin, A., & Brezinski, K. L. (2000). Reliability methods. *Educational and Psychological Measurement*, 60, 523–531.
- Iacobucci, D. & Duchachek, A. (2003). Advancing alpha: measuring reliability with confidence. *Journal of Consumer Psychology*, 13, 478-487.
- Koning, A. J. & Franses, P. H. (2003, June). *Confidence intervals for Cronbach's Coefficient Alpha values*. ERIM Report Series Reference No. ERS-2003-041-MKT.
- Kristof, W. (1963). The statistical theory of stepped-up reliability coefficients when a test has been divided into several equivalent parts. *Psychometrika*, 28, 221–238.
- Lord, F. M. (1974). Variance stabilizing transformation of the stepped-up reliability coefficient. *Journal of Educational Measurement*, 11, 55–57.
- Maydeu-Olivares, A., Coffman, D. L., & Hartmann, W. M. (2007) Asymptotically distribution-free (ADF) interval estimation of coefficient alpha. *Psychological Methods*, 12, 157-176.
- Peterson, R. A. (1994). A meta-analysis of Cronbach's coefficient alpha. *Journal of Consumer Research*, 21, 381–391.
- Robey, R., & Barciowski, R. (1992) Type I error and the number of iterations in Monte Carlo studies of robustness. *British Journal of Mathematics and Statistical Psychology* 45, 283 –288.
- Romano, J. L., Kromrey, J. D., & Hibbard, S. (2008, March). *Confidence interval methods for coefficient alpha: A Monte Carlo study of methods proposed in the literature*. Paper presented at the annual meeting of the American Educational Research Association, New York.
- Schafer, J. L. & Graham, J.W. (2002). Missing data: Our view of the state of the art. *Psychological Methods*, 7, 147-177.
- Sinharay, S., Stern, H.S., & Russell, D. (2001). The use of multiple imputation for the analysis of missing data. *Psychological Methods*, 6, 317-329.

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

Your comments and questions are valued and encouraged. Please contact Jeff Kromrey at:
University of South Florida
4202 East Fowler Ave. EDU 162
Tampa, FL 33620
Work Phone: 813-974-5739
Fax: 813-974-4495
Email: kromrey@tempest.coedu.usf.edu