Using SAS® to Calculate Kappa and Confidence Intervals for Binary Data with Multiple Raters, and for the Consensus of Multiple Diagnoses

Art Noda, Stanford University School of Medicine, Stanford, CA
Helena Chmura Kraemer, Stanford University School of Medicine, Stanford, CA
Vyjeyanthi S. Periyakoil, VA Palo Alto Health Care System, Palo Alto, CA

ABSTRACT

Suppose you go to the doctor to get a medical diagnosis. How confident would you be of that diagnosis? If you could get a second, third, or even fourth opinion, how many positive diagnoses would convince you, and how reliable would that consensus be?

If each of a sample of patients are evaluated by enough raters and are independently diagnosed as either positive or negative, we can evaluate the reliability (kappa coefficient and corresponding confidence interval), of each consensus of 2, 3, 4, ..., M raters. We can select the optimal consensus, and demonstrate an increase in reliability with multiple diagnoses. Results indicate that the majority rule does not always work, nor does any single rule, which leaves determination of the optimal consensus to empirical evaluation.

SAS code is presented that calculates the kappa coefficient and the confidence intervals (using a jackknife technique) for these cases and determines the optimal consensus. This code requires a dataset containing 2 categories, at least M=4 raters, and at least N=20 patients (items).

INTRODUCTION

The kappa coefficient (k), introduced for M=2 raters by Cohen (1960), was estimated for the 2XM intraclass kappa (2 categories, M raters) case by Fleiss (1981). There are several ways to calculate this statistic, but the easiest both for theory and application requires that the data be organized by s, the number of positive diagnoses per patient; s = 0, 1, 2, ..., M. The proportion of the N patients sampled who have s positive diagnoses is f_s. Then the sample kappa, k, is:

\[ k = 1 - \frac{\sum_{s=0}^{M} f_s \cdot \left( \frac{s}{M} \right) \left( 1 - \frac{s}{M} \right)}{(M - 1)P(1 - P)} \]

where

\[ P = \sum_{s=0}^{M} f_s \cdot \left( \frac{s}{M} \right). \]

Note that when there are M=2 raters, the simple and weighted kappas and corresponding confidence intervals can be obtained using the AGREE option in PROC FREQ. Also, SAS Institute supplies a macro on its website (http://ftp.sas.com/techsup/download/stat/magree.html) that calculates the simple and weighted kappa when the number of raters is greater than two, but not the confidence intervals. Finally, Vierkant (1997) provides a SAS macro which calculates bootstrapped confidence intervals for both simple and weighted kappa when there are 2 raters. These programs may be useful for your situation.

However, to date, when M>2 the standard error is easily accessible only under the hypothesis of randomness (Fleiss 1981). An approximate standard error of kappa can be calculated using Jackknife procedures omitting one patient (subject) at a time (Bloch and Kraemer, 1989). For this paper, we shall use the jackknife to calculate both a jackknife kappa and its confidence interval. The jackknife kappa is defined as follows:

\[ J_s = \text{Jackknife } k_s = Nk - (N - 1)k_s \]

where \( k_s \) is the kappa coefficient based on omitting one subject with s positive diagnoses, and \( f_s \) is the proportion of such subjects among the N total subjects.

The 95% confidence intervals are calculated as follows:

\[ 95\% \text{ CI} = \bar{J} \pm 1.96 \times SD / \sqrt{N} \]

where

\[ SD^2 = \sum_s \left( f_s (J_s - \bar{J})^2 / (\sum_s f_s - 1) \right). \]

Jackknife results correspond closely to those derived in various ways for the 2X2 intraclass kappas (Kraemer, 2000, submitted). As a "rule of thumb", the minimum number of patients should exceed both 10/P, and 10/(1-P), where P is the estimated proportion of positive diagnoses. For example, when P=0.5, 20 patients are minimal; when P=0.01, no fewer than 1000 patients are needed.

Landis and Koch (1977) interpret kappa as follows: k <= 0 is defined as "poor", 0 to .2 "slight", .2 -.4 "fair", .4 -
A context of medical research in which kappa coefficients have proved uniquely useful is that of the consensus diagnosis (Kraemer, 2000, submitted). Suppose one assesses the reliability of a single binary $X_{ij}$ and found that its reliability, as measured by an intraclass kappa, was greater than zero, but not satisfactory. Could one use a consensus of $M$ raters, requiring at least $C$ positive diagnosis for a consensus positive diagnosis, and thereby achieve adequate reliability by the above standards? How large should $M$ be, and what value of $C$ should be chosen?

**METHOD**

For a consensus of $M$ raters, sample at least $KM$ raters for each patient sampled. Randomly split the raters into $K$ groups of $M$ for each patient. Then for $C = 1, 2, \ldots, M$, defining the consensus diagnosis, evaluate the $2XK$ intraclass kappa: $k_{CM}$. Choose the optimal cutpoint $C$ as the one that maximizes $k_{CM}$.

Note that if the optimal consensus of 2 is obtained when $C=1$, in practice one would request a second opinion when the first diagnosis was positive, if the optimal consensus of 2 is obtained when $C=2$, in practice one would not request a second opinion when the first one was negative. It often happens with the optimal consensus that, when put into practice, the number of ratings per patient to obtain a consensus of $M$ is far less than $M$ ratings per patient. This often means one can increase the quality of the diagnosis with minimal increase in time and cost. However, to identify that optimal consensus in the first place requires at least $2M$ ratings per patient. Thus, to evaluate a consensus of 3, one needs 6 ratings per patient, for 4, one needs 8, etc. The SAS code in the Appendix is designed for 12 raters (with 2 raters not used in the calculation) and 2 groups of 6 raters, and the corresponding kappa values are calculated for $C=1, 2, \ldots, M$ raters.

**RESULTS**

The results are presented in Table 1 and graphed in Figure 1. “P” is the percentage of positive consensus diagnoses.

**Table 1.** Kappa, Jackknife Kappa, and CI's for Periyakoil Data

<table>
<thead>
<tr>
<th># of</th>
<th>J.K. 95%</th>
<th>Upper 95%</th>
<th>Lower 95%</th>
</tr>
</thead>
<tbody>
<tr>
<td>M C Groups</td>
<td>P</td>
<td>Kappa</td>
<td>Kappa</td>
</tr>
<tr>
<td>1 1</td>
<td>12</td>
<td>0.52</td>
<td>0.42</td>
</tr>
<tr>
<td>2 1</td>
<td>6</td>
<td>0.66</td>
<td>0.59</td>
</tr>
<tr>
<td>2 2</td>
<td>6</td>
<td>0.39</td>
<td>0.34</td>
</tr>
<tr>
<td>3 1</td>
<td>4</td>
<td>0.74</td>
<td>0.65</td>
</tr>
<tr>
<td>3 2</td>
<td>4</td>
<td>0.54</td>
<td>0.63</td>
</tr>
<tr>
<td>3 3</td>
<td>4</td>
<td>0.30</td>
<td>0.31</td>
</tr>
<tr>
<td>4 1</td>
<td>3</td>
<td>0.75</td>
<td>0.64</td>
</tr>
<tr>
<td>4 2</td>
<td>3</td>
<td>0.64</td>
<td>0.71</td>
</tr>
<tr>
<td>4 3</td>
<td>3</td>
<td>0.47</td>
<td>0.55</td>
</tr>
<tr>
<td>4 4</td>
<td>3</td>
<td>0.23</td>
<td>0.26</td>
</tr>
<tr>
<td>5 1</td>
<td>2</td>
<td>0.78</td>
<td>0.71</td>
</tr>
<tr>
<td>5 2</td>
<td>2</td>
<td>0.67</td>
<td>0.70</td>
</tr>
<tr>
<td>5 3</td>
<td>2</td>
<td>0.57</td>
<td>0.82</td>
</tr>
<tr>
<td>5 4</td>
<td>2</td>
<td>0.41</td>
<td>0.46</td>
</tr>
<tr>
<td>5 5</td>
<td>2</td>
<td>0.22</td>
<td>0.15</td>
</tr>
<tr>
<td>6 1</td>
<td>2</td>
<td>0.81</td>
<td>0.62</td>
</tr>
<tr>
<td>6 2</td>
<td>2</td>
<td>0.70</td>
<td>0.86</td>
</tr>
<tr>
<td>6 3</td>
<td>2</td>
<td>0.59</td>
<td>0.76</td>
</tr>
<tr>
<td>6 4</td>
<td>2</td>
<td>0.51</td>
<td>0.71</td>
</tr>
<tr>
<td>6 5</td>
<td>2</td>
<td>0.38</td>
<td>0.45</td>
</tr>
<tr>
<td>6 6</td>
<td>2</td>
<td>0.15</td>
<td>0.27</td>
</tr>
</tbody>
</table>

**Figure 1.** Jackknife Kappa Values vs. Percentage of Positive Consensus Diagnoses (P)

We then randomly split the pool of 12 raters into six sets of two randomly selected raters, and found that $k_{12}=0.34$ and $k_{22}=0.59$. Thus the optimal consensus of 2 is to use a cutpoint $C=1$ as the reliability then rises from $k_{11}=0.42$ to $k_{22}=0.59$. The pool of 12 raters is similarly split into 3 groups of 4 raters, 4 groups of 3 raters, 2 groups of 5 raters (with 2 raters not used in the calculation) and 2 groups of 6 raters, and the corresponding kappa values are calculated for $C=1, 2, \ldots, M$ raters.

**EXAMPLE**

In a study conducted by Periyakoil (Kraemer, 2000, submitted), N=69 items were sampled from the population of items that might be used to distinguish preparatory grief (positive diagnosis) from depression (negative diagnosis) in dying adult patients. The issue was to assess to what extent clinicians could reliably distinguish between the two. $M=12$ expert clinicians were sampled and were asked to classify each item as more indicative of preparatory grief or depression. When a clinician did not respond to an item, it was randomly assigned either grief or depression with a probability of .5 for each. The resulting kappa ($k_{11}$) for a single rating (2X12 kappa) was 0.42, considered to be "moderate". 

A "moderate", .6 - .8 "substantial", .8 -1 "almost perfect". Kappa equal to 1 denotes perfect agreement.

$\frac{.6 \text{ "moderate"}, .6 - .8 \text{ "substantial"}, .8 - 1 \text{ "almost perfect". }}{0.6}$
To obtain a jackknife kappa $\geq .8$ (almost perfect), one minimally needs two groups of 5 raters, requiring at least 3 positive diagnoses for a consensus. Note also that the lower 95% CI for this case (.69) is greater than the upper 95% CI for the $M=1$ rater case (.51), indicating that an improvement in kappa has been achieved.

CONCLUSION

By using a consensus of M raters, requiring at least C positive diagnoses for a consensus positive diagnosis, a higher level of reliability may be achieved. For the Peryakoil data, the jackknife kappa coefficient increased from 0.42 (moderate) to 0.87 (almost perfect) as M increased from 1 to 6. If you have 2 categories, at least M=4 raters, and at least N=20 patients in your dataset, by appropriately modifying the SAS program in the Appendix, the optimal C and M can be estimated for your data.

REFERENCES


APPENDIX

* This program calculates kappa, the jackknife kappa, and 95% CIs for the jackknife kappa for all possible consensus diagnosis combinations.

To successfully execute this program, the number of categories, K, must equal 2, and the number of raters, M, must be at least 4. While this program analyzes the Periyakoil data with $M=12$ raters and $N=69$ subjects, the code may be easily modified for any number of raters greater than or equal to 4 (though it will become unwieldy for a large number of raters). The number of subjects (patients) should be at least 20.

The data to be analyzed (RAWDATA) must be in the following format, where 1=Positive Diagnosis, 0=Negative Diagnosis

```
N Rater1 Rater2 ... RaterM
--- ------ ... -------
1 0 1 ... 1
2 1 1 ... 0
... ... ... ...
N 0 0 ... 1
```

* Enter number of subjects;
* Enter number of raters;
* Number of categories (Must equal 2, consisting of 1's and 0's);

```
%let n=69;
%let m=12;
%let k=2;
```

* Transpose raw data;
```
proc transpose data=rawdata out=tfill;
var rater1-rater&m;
run;
```

* Assign a random number to each rater's response and sort. This randomizes responses among raters. Note that '816' was randomly selected as a seed;
```
%macro randsort;
%do i=1 %to &n;
data item&i; set tfill;
    rand&i=ranuni(816*&i);
    keep _name_ rand&i col&i;
run;
proc sort data=item&i; by rand&i;
run;
```

* Merge together the randomized responses and
```
data item&i; set item&i;
    rename col&i=rndrat&i;
    keep col&i;
run;
%mend randsort;
```

CONTACT INFORMATION

Art Noda
Stanford University School of Medicine
Department of Psychiatry, 5550
Stanford, CA 94305
artnoda@stanford.edu
transpose data back to original format.
The responses for the 12 "raters" have now been randomized. Rename variables as RNDRAT1 to RNDRAT12;
%macro names(number);
  %do i=1 %to &n;
    item\&i
  %end;
%mend names;
data trandom;
merge %names(&n); run;
proc transpose data=trandom out=random; var RNDRAT1··RNDRAT&n; run;
%macro rename(number);
  %do i=1 %to &n;
    col\&i=rndrat\&i
  %end;
%mend rename;
data random;
set random;
rename %rename(&m);
drop _name_; run;
* Rename RNDRATxx variables in same format as consensus diagnosis variables described below;
%macro consen11(number);
  %do i=1 %to &n;
    if rndrat\&i=1 then C1_2_\&i=1;
    else C1_2_\&i=0;
  %end;
%mend consen11;
* Consider the first two random raters. If at least one of the two random raters gives
a positive diagnosis, we consider the results to be a consensus diagnosis.
If this is the case, assign a "1" to a new variable, C1_2_\&i, where "_2" in the
variable name indicates at least 1 out of 2 raters is necessary to achieve a consensus diagnosis.
We will have a total of 6 new variables, (C1_2_1, C1_2_3, C1_2_5, C1_2_7, C1_2_9,
and C1_2_11) each with the results of 2 random raters, so the data from all 12 raters is used;
%macro consen12(number);
  %do i=1 %to &n-1 %by 2;
    %do j=\&i %to \&i+1;
    if sum(of rndrat\&i -- rndrat\&j) ge 2
      then C2_2_\&i=1;
      else C2_2_\&i=0;
    %end;
%mend consen12;
* Here, we require both random raters to give a positive diagnosis to achieve a consensus diagnosis (C2_2_\&i=1). Otherwise, assign a zero to these six variables;
%macro consen22(number);
  %do i=1 %to \&n-1 %by 2;
    %do j=\&i %to \&i+1;
      if sum(of rndrat\&i -- rndrat\&j) ge 2
        then C2_2_\&i=1;
        else C2_2_\&i=0;
    %end;
%mend consen22;
* Now we take 3 random raters at a time and require one of the three to give a positive
diagnosis to achieve a consensus diagnosis (C3_3_\&i=1). Otherwise assign a zero. A
total of four variables are created. (C1_3_1, C1_3_4, C1_3_7, C1_3_10);
%macro consen13(number);
  %do i=1 %to \&n-2 %by 3;
    %do j=\&i+1 %to \&i+1;
    if sum(of rndrat\&i -- rndrat\&j) ge 2
      then C3_3_\&i=1;
      else C3_3_\&i=0;
    %end;
%mend consen13;
* Now, 2 of 3 random raters are needed to give a positive diagnosis to achieve
a consensus diagnosis;
%macro consen23(number);
  %do i=1 %to \&n-2 %by 3;
    %do j=\&i+1 %to \&i+1;
    if sum(of rndrat\&i -- rndrat\&j) ge 2
      then C2_3_\&i=1;
      else C2_3_\&i=0;
    %end;
%mend consen23;
* All 3 random raters are needed to give a positive diagnosis to achieve
a consensus diagnosis;
%macro consen33(number);
  %do i=1 %to \&n-2 %by 3;
    %do j=\&i+1 %to \&i+1;
    if sum(of rndrat\&i -- rndrat\&j) ge 3
      then C3_3_\&i=1;
      else C3_3_\&i=0;
    %end;
%mend consen33;
* Repeat above code for 4 random raters (1, 2, 3, and 4 positive diagnoses for a consensus
diagnosis), 5 random raters (in this case, 2 random raters are leftover and not used),
and 6 random raters. This code is not included due to space limitations, but can be
easily created using the code above as a template;
* Execute the CONSENxx macros created above
Add N, M, K, and subject number to the data;
data random&dsnum; set random;
  keep subject n k
c&cnum1._&cnum2._1··
c&cnum1._&cnum2._&lnum;
run;
data init&dsnum; set random&dsnum;
  group=&group;
cat_n = sum(of c&cnum1._&cnum2._1··
c&cnum1._&cnum2._&lnum);
cat_p = sum(of c&cnum1._&cnum2._1··
c&cnum1._&cnum2._&lnum);
p_n=cat_n/group;
p_p=cat_p/group;
max_1=max(cat_n,cat_p);
min_1=min(cat_n,cat_p);
keep subject
c&cnum1._&cnum2._1··
c&cnum1._&cnum2._&lnum cat_n cat_p
p_n p_p n group k max_1 min_1;
run;
proc sort data=init&dsnum;
  by descending max_1 descending cat_n descending cat_p;
run;
proc print data=init&dsnum
  format p_n .p_p 4.2;
  title "Data &dsnum:.For Consensus Diagnosis";
run;
data init&dsnum;
  set init&dsnum;
  mergevar=1;
  keep c&cnum1._&cnum2._1··
c&cnum1._&cnum2._&lnum cat_n cat_p
p_n p_p n group k max_1 min_1 mergevar;
run;
proc means data=init&dsnum noprint nway;
  var t;
  output out=sum_t&dsnum sum=sum_t;
run;
data sum_t&dsnum; set sum_t&dsnum;
  mergevar=1;
  drop _type_ _freq_;run;
data init&dsnum;
  merge init&dsnum sum_t&dsnum;
  by mergevar;
n=group*sum_t;
den=(group-1)*(mn_pq_n+mn_pq_p);
kappa=1-(num/den);
ki=1-(((n-1)*(num-(group*t)))/
  ((n*den)-(n*(group-1)*w)+(group-1)*t));
ji=(n*kappa) - (((n-1)*ki);
run;
proc freq data=init&dsnum noprint;
  tables cat_p / out=freq_p&dsnum;
run;
data freq_p&dsnum;
  set freq_p&dsnum;
  s_div_m=s/max_s;
p=fs * s_div_m;
run;
proc means data=freq_p&dsnum noprint nway;
  var s;
  output out=s&dsnum max=max_s;
run;
data s&dsnum; set s&dsnum;
  mergevar=1;
  keep max_s mergevar;
run;
data freq_p&dsnum;
  merge freq_p&dsnum s&dsnum;
  by mergevar;
run;
data freq_p&dsnum;
  set freq_p&dsnum;
  s_div_m=s/max_s;
p=fs * s_div_m;
run;
proc means data=freq_p&dsnum noprint nway;
var p;
output out=p\&dsnum sum=p;
run;
data p\&dsnum; set p\&dsnum;
mergevar=1;
keep mergevar p;
run;
proc means data=init\&dsnum noprint nway;
  var ji;
  output out=stat_j\&dsnum mean=j std=sd_j;
run;
data stat_j\&dsnum; set stat_j\&dsnum;
mergevar=1;
drop _type_ _freq_; 
run;
data kappa\&dsnum; 
  merge init\&dsnum stat_j\&dsnum p\&dsnum;
  by mergevar;
run;
data kappa\&dsnum; set kappa\&dsnum;
  se_j=sd_j/sqrt(n);
  j_2t_95u=j+1.96*se_j; j_2t_95l=j-1.96*se_j;
run;
data kappa\&dsnum; set kappa\&dsnum;
  by mergevar;
  dataset=&dsnum;
  if last.mergevar;
run;
proc print data=kappa\&dsnum;
format dataset n group k kappa j j_2t_95l j_2t_95u 6.3;
var dataset n group k kappa j j_2t_95l j_2t_95u;
title 'Data Set &dsnum -- Kappa&dsnum calcs';
run;
%calc;
%calc(11,1,12,12); %calc(12,1,12,11, 6);
%calc(22,2,11, 6); %calc(13,1,3,10, 4);
%calc(23,2,3,10, 4); %calc(33,3,3,10, 4);
%calc(14,1,4, 9, 3); %calc(24,2,4, 9, 3);
%calc(34,3,4, 9, 3); %calc(44,4,4, 9, 3);
%calc(15,1,5, 6, 2); %calc(25,2,5, 6, 2);
%calc(35,3,5, 6, 2); %calc(45,4,5, 6, 2);
%calc(55,5,5, 6, 2); %calc(16,1,6, 7, 2);
%calc(26,2,6, 7, 2); %calc(36,3,6, 7, 2);
%calc(46,4,6, 7, 2); %calc(56,5,6, 7, 2);
%calc(66,6,6, 7, 2);
* Combine all calculated kappa, jackknife kappa, and 95% CI results and print;
data kappasall;
set kappa11 kappa12 kappa22 kappa13 kappa23 kappa33 kappa14 kappa24 kappa34 kappa44 kappa15 kappa25 kappa35 kappa45 kappa55 kappa16 kappa26 kappa36 kappa46 kappa56 kappa66;
run;
proc print data=kappasall;
format kappa j j_2t_95l j_2t_95u 6.3;
var dataset n k group p kappa j j_2t_95l j_2t_95u;
title 'Kappa, jackknife kappa and 95% CIs';
run;
%end calc;