Paper 208-28

Analysis of Method Comparison Studies Using SAS®

Mohamed Shoukri, King Faisal Specialist Hospital & Research Center, Riyadh, KSA and Department of Epidemiology and Biostatistics, University of Western Ontario, London, Ontario, Canada
Samia Hashim, King Faisal Specialist Hospital & Research Center, Riyadh, KSA

ABSTRACT

Method comparison studies are used to assess the relative agreement between two analytical methods that measure the same quantity. The primary objective is to see whether two methods produce the same errors in measurement.

The aim of this poster is to present a SAS source code for analyzing a method comparison study. For the purpose of demonstration, data on bacterial counts from nasal swabs taken from cattle and measured by two clinicians were analyzed to assess the interclinicians agreement.

The SAS source code makes use of the following: the TRANSPOSE procedure, the CORR procedure, the MEANS procedure, the GPLOT procedure and macro language.

The analytic techniques presented can be used on any operating system and are intended for an intermediate level audience.

KEYWORDS

PROC TRANSPOSE, PROC CORR, PROC MEANS, PROC GPLOT, Macro Language, Mean, Standard Deviation, Vertical Reference.

INTRODUCTION

Comparison of two methods of measuring some quantity is very common. In medicine, one may need to compare two methods of measurements, such as, blood pressure levels, estimation of gestational age, or weights of newborn infants. In clinical chemistry, methods of comparison of two assays are of common interest. When two methods are compared, neither one can be regarded as giving a true value of the quantity being measured. In this case it is important to know whether the methods give measurements that are in some sense comparable. For example, lab technologists may wish to see whether a new, inexpensive and quick method produces measurements that are clinically in close agreement with those from an established method. Using the t-test or the Pearson's product-moment correlation is flawed. Problems with using the familiar correlation analysis and t-test have been noted by Altman and Bland (1983) and Bland and Altman (1986). In their papers they proposed several techniques for analyzing method comparison studies and many articles advocating their approach or variations of it have since appeared in many journals including the "Annals of Clinical Biochemistry" (1996). The Bland and Altman's approach with the framework of linear error measurement model will be discussed, but first let us assume that each method measures the same set of subjects and that the data layout takes the form:

	Subject				
	1	2	3	………………….	k
Method (1) (X)	x_1	x_2	x_3	………………….	x_k
Method (2) (Y)	y_1	y_2	y_3	………………….	y_k

The basic idea of Bland and Altman's approach is the examination of the difference of the readings made by the two methods, then plotting the difference $v_i = (x_i - y_i)$ against the mean $u_i = \dfrac{x_i + y_i}{2}$ for each subject. From this plot, it is easy to see if there is any tendency for the variation to change with the magnitude of the measurements. If the differences are symmetrical around zero, then there is no systematic bias. If there is no relationship between the differences and the averages, the agreement between the two methods may be summarized using the means and standard deviations of the methods.

This poster has two-fold objectives:

(1) Test the hypothesis that the two methods are equally precise using Shukla's (1973) method. Shukla proved that this hypothesis is tested by referring $t = \dfrac{r\sqrt{k-2}}{\sqrt{1-r^2}}$ to the table of ***t*** distribution with k-2 degrees of freedom where $r = correlation$ between (v,u) .

(2) Plotting v against u , with the
Upper Limit $(v) = mean(v) + 2\ \ SD(v)$
Lower Limit $(v) = mean(v) - 2\ \ SD(v)$

on the same graph.

In the following sections, we thoroughly discuss the procedures and steps used in the SAS source code provided in the Appendix to achieve the above objectives.

THE DATA

Clinicians from the Ontario Veterinary College were asked to read bacterial count on 15 slides. Each clinician was asked to read each slide twice (blindly). Therefore, each slide has 4 readings and the main objectives of the experiment were:

(1) To graphically assess the extent of agreement between the two clinicians using Bland and Altman's approach.

(2) To use Shukla's approach to test the hypothesis that the two clinicians are equally precise in their readings.
The data set **`rater`**, shown below, consists mainly of four variables: `slide`, `rater`, `reading` and `count`. Where:

`slide` = identifier
`rater` = clinician
`reading` = 2 readings per rater
`count` = variable of interest

The data set **rater**:

```
data rater;
  input slide rater reading count;
  datalines;
  1 1 1  52
  1 1 2  58
  1 2 1  70
  1 2 2  72
  2 1 1  68
  2 1 2  65
  2 2 1  68
  2 2 2  68
  3 1 1 106
  3 1 2 104
  3 2 1 102
  3 2 2 101
  4 1 1  98
  4 1 2  98
  4 2 1  95
  4 2 2  98
  5 1 1  40
  5 1 2  45
  5 2 1  40
  5 2 2  44
  6 1 1  87
  6 1 2  88
  6 2 1  90
  6 2 2  88
  7 1 1  98
  7 1 2 105
  7 2 1  89
  7 2 2 104
  8 1 1  98
  8 1 2 101
  8 2 1  99
  8 2 2 100
  9 1 1 122
  9 1 2 121
  9 2 1 118
  9 2 2 118
 10 1 1  75
 10 1 2  76
 10 2 1  71
 10 2 2  71
 11 1 1  80
 11 1 2  83
 11 2 1  82
 11 2 2  80
 12 1 1  94
 12 1 2  95
 12 2 1  98
 12 2 2  95
 13 1 1  98
 13 1 2  98
 13 2 1  98
 13 2 2  94
 14 1 1 105
 14 1 2 111
 14 2 1 111
 14 2 2 107
 15 1 1  93
 15 1 2 105
 15 2 1 101
 15 2 2 104
  ;
run;
```

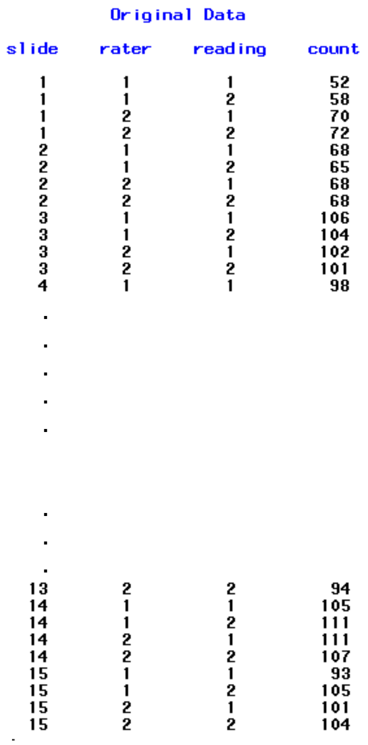
THE TRANSPOSE PROCEDURE

It is necessary to transpose the data from the univariate structure into a multivariate structure to facilitate the computation of the difference and the sum of the readings made by the raters. As well, this step is required to compute the Pearson's Correlation Coefficient between the sum and the difference of the averages of the readings for each slide to test the equality of the precisions of the two clinicians.

In order to see what the transposed data look like, let us first run the original data by using PROC PRINT:

```
options nodate nonumber;
proc print data=rater noobs;
  title 'Original Data';
run;
```

The following is a partial listing of the output



```
             Original Data

slide    rater    reading    count

  1        1         1         52
  1        1         2         58
  1        2         1         70
  1        2         2         72
  2        1         1         68
  2        1         2         65
  2        2         1         68
  2        2         2         68
  3        1         1        106
  3        1         2        104
  3        2         1        102
  3        2         2        101
  4        1         1         98
  .
  .
  .
  .
  .


  .
  .
  .
 13        2         2         94
 14        1         1        105
 14        1         2        111
 14        2         1        111
 14        2         2        107
 15        1         1         93
 15        1         2        105
 15        2         1        101
 15        2         2        104
```

The TRANSPOSE procedure:

```
* Transpose the original data from its
  univariate structure to a multivariate
  structure;
proc transpose data=rater
               out=ratertrans
               (rename=(rater1=rater11
                        rater2=rater12
                        rater3=rater21
                        rater4=rater22)
               drop=_NAME_)
               prefix=rater;
  var count;
  by  slide;
run;
```

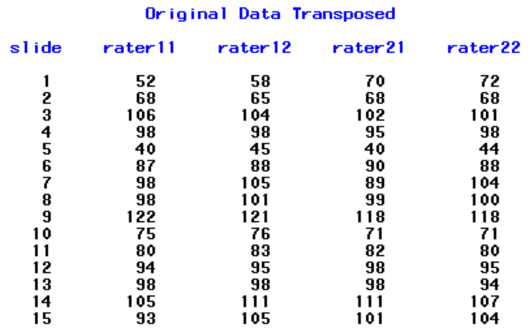
The TRANSPOSE procedure reads in the data set **rater** then saves the transposed version in a new data set called **ratertrans**. The BY statement generates an observation for each variable being transposed **(count)** for each BY group

(slide). The PREFIX= option of the PROC TRANSPOSE statement specifies the prefix to use in constructing names of transposed variables in the output data set **(rater1, rater2, rater3, rater4)**. For the purpose of recording different readings done by the same rater, we rename the transposed variable names to **rater11, rater12, rater21** and **rater22** consecutively. _NAME_ is an automatic variable that is created in the output data set **(ratertrans)**. It contains the name of the variable being transposed **(count)** to create the current observation, and since there is no need to keep track of the transposed variable **(count)**, the **_NAME_** variable is therefore dropped.

To see what the transposed data look like, we do a PROC PRINT:

```
proc print data=ratertrans noobs;
  title 'Original Data Transposed';
run;
```

When run, the following output is generated:



```
               Original Data Transposed

slide     rater11     rater12     rater21     rater22

   1         52          58          70          72
   2         68          65          68          68
   3        106         104         102         101
   4         98          98          95          98
   5         40          45          40          44
   6         87          88          90          88
   7         98         105          89         104
   8         98         101          99         100
   9        122         121         118         118
  10         75          76          71          71
  11         80          83          82          80
  12         94          95          98          95
  13         98          98          98          94
  14        105         111         111         107
  15         93         105         101         104
```

The DATA step, shown below, reads in the data set **ratertrans** and creates two new variables x and y that compute the average of the readings made by each rater. To implement Bland and Altman's approach, we need to evaluate the sum of the averages of the readings which is denoted by u and the difference of the averages of the readings which is denoted by v resulting in the data set **newrater**. Bland and Altman's approach calls for plotting v versus u on the same plot. We need to add an upper and lower limit (v) where:

Upper Limit $(v) = mean(v) + 2 \;\; SD(v)$

Lower Limit $(v) = mean(v) - 2 \;\; SD(v)$

The data set **newrater**:

```
data newrater;
  set ratertrans;
* 1. Compute the averages of the readings
     made by each rater;
      x = (rater11 + rater12)/2;
      y = (rater21 + rater22)/2;
* 2. Compute the sum and the difference
     of the averages of the readings;
      u = x + y;
      v = x - y;
run;
```

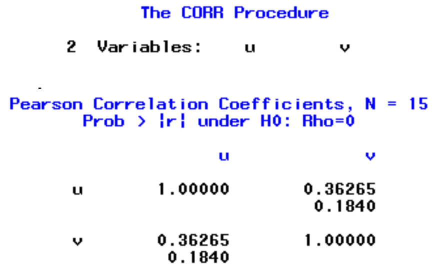
THE CORR PROCEDURE

The CORR procedure allows us to measure the correlation between (v,u). If this correlation is not significantly different from zero then according to Shukla's approach, the two clinicians are declared to be equally precise. Therefore the $p-value$ that appears below the value of the reported Pearson's Correlation Coefficient is the tool to assess the equality of the precisions of the two clinicians.

The CORR procedure:

```
* Measure the correlation between u and v;
proc corr data=newrater;
  var u v;
  title;
run;
```

Results from the CORR procedure:



```
                The CORR Procedure

   2  Variables:    u        v


Pearson Correlation Coefficients, N = 15
        Prob > |r| under H0: Rho=0

                    u              v

   u         1.00000          0.36265
                               0.1840

   v         0.36265          1.00000
              0.1840
```

From the above results, the Pearson's Correlation Coefficient is 0.362 and the $p-value$ is 0.184, and since the $p-value$ is greater than 0.05 then we should accept the hypothesis that the two raters are equally precise.

THE MEANS PROCEDURE

In order to calculate the upper and lower limits (v), we need to retrieve the value of the mean and the standard deviation of v in a variable format. Therefore we do a PROC MEANS on v using the MEAN and STD options of the OUTPUT statement as shown below:

```
* Retrieve the value of the mean
  and the standard deviation of v;
proc means data=newrater;
  var v;
  output mean=meanv std=sdv out=reflines;
run;
```

The DATA step, below, creates the data set **lines** that reads in the data set **reflines** in order to compute both the upper and lower limits (v). The variables, **up** and **lo**, must be converted into macro variables since the VREF= option in the GPLOT procedure can only accept numeric constants.

To convert variables into macro variables in a DATA step, we use a macro statement called CALL SYMPUT as shown below:

```
data lines;
  set reflines;
* 1. Compute the upper and lower limits of v;
      up = meanv + (2*sdv);
      lo = meanv - (2*sdv);
* 2. Convert the variables, up and lo, into
     macro variables;
      call symput('upper',up);
      call symput('lower',lo);
run;
```

Where the current value of the variable name **up** is assigned to the macro variable **'upper'** and likewise for variable name **lo**. Whenever a macro variable needs to be referenced from anywhere within a program, we type the macro variable name without single quotes preceded by an ampersand (&) sign, for example, **&upper**.

THE GPLOT PROCEDURE

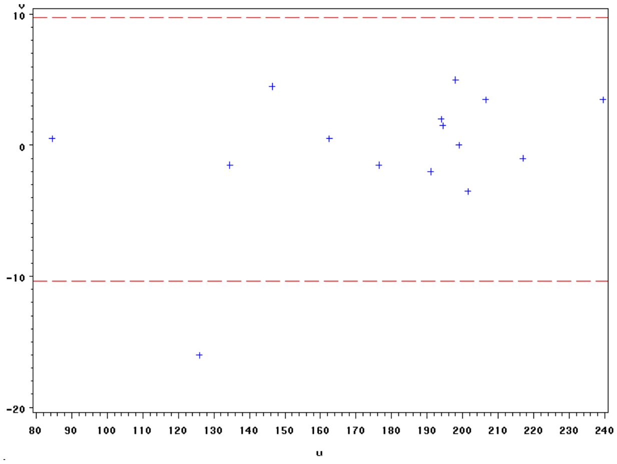
The GPLOT procedure produces a scatter plot of v against u with the confidence limits of v drawn on the same graph.

The GPLOT procedure, shown below, makes use of the VREF= option which assigns the variables, **upper** and **lower**, to it; each preceded by the ampersand (&) symbol in order to reference the macro variables created earlier. The VREF= option draws two reference lines perpendicular to the vertical axis at points specified by the values assigned to it (**&upper** and **&lower**). The LVREF= option specifies the line type for drawing reference lines requested by the VREF= option. There are 46 predefined line types to select from in SAS/GRAPH®. By default, line type 2 is set for confidence limit lines. The CVREF= option specifies the color for reference lines requested by the VREF= option.

The GPLOT procedure:

```
   * Produce a scatter plot with the
     upper and lower limits of v;
   goptions reset=all;
   symbol1 v=plus c=blue;
   proc gplot data=newrater;
     plot v*u / vref=&upper &lower lvref=2
                cvref=red;
   run;
   quit;
```

When the GPLOT procedure is run, the following scatter plot graph is produced:



CONCLUSION

From the scatter plot, a general picture may emerge. Points that fall outside the limits of v maybe considered outliers. Moreover, if there is no apparent trend in the scatter plot then there is no relationship between v and u, and this indicates reasonable agreement between the two clinicians. However, if there is a relationship between v and u depicted from the plot then, as Bland and Altman suggested, we may employ a transformation on the original data and repeat the entire procedure. In many situations, the logarithmic transformation may suffice.

REFERENCES

Altman, D.G. and Bland, J.M. (1983), “Measurement in Medicine: The Analysis of Method Comparison Studies,” *The Statistician*, 32, 307-317.

Bland, J.M. and Altman, D.G. (1986), “Statistical Methods for Assessing Agreement between Two Methods of Clinical Measurement,” *Lancet*, 8, 307-310.

Hollis, S. (1996), “Analysis of Method Comparison Studies,” *Annals of Clinical Biochemistry*, 33, 1-4.

Shukla, G.K. (1973), “Some Exact Tests of Hypotheses About Grubbs's Estimators,” *Biometrics*, 29, 373-377.

SAS Institute Inc. (1990), *SAS Procedure Guide, Version 6, Third Edition*, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1990), *SAS/GRAPH Software: Reference, Version 6, First Edition, Volume 1*, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1990), *SAS/GRAPH Software: Reference, Version 6, First Edition, Volume 2*, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1989), *SAS/STAT User’s Guide, Version 6, Fourth Edition, Volume 2*, Cary, NC: SAS Institute Inc.

Burlew, Michele M. (2001), *SAS Macro Programming Made Easy, 280 Pages*, Cary, NC: SAS Institute Inc.

ACKNOWLEDGEMENT

The authors would like to thank Michele Ensor for reviewing this poster and making valuable suggestions.

CONTACT INFORMATION

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

Samia Hashim
King Faisal Specialist Hospital & Research Center
Biostatistics, Epidemiology & Scientific Computing Department
MBC 03-BESC
PO Box 3554
Riyadh 11211
KSA
Work Phone: (+9661) 464-7272 Ext. 39229
Fax: (+9661) 442-4542
Email: samia@kfshrc.edu.sa
Website: www.kfshrc.edu.sa

Mohamed Shoukri
King Faisal Specialist Hospital & Research Center
Biostatistics, Epidemiology & Scientific Computing Department
MBC 03-BESC
PO Box 3554
Riyadh 11211
KSA
Work Phone: (+9661) 464-7272 Ext. 32526
Fax: (+9661) 442-4542
Email: shoukri@kfshrc.edu.sa
Website: www.kfshrc.edu.sa

APPENDIX

The SAS code:

```
data rater;
  input slide rater reading count;
  datalines;
  1  1 1  52
  1  1 2  58
  1  2 1  70
  1  2 2  72
  2  1 1  68
  2  1 2  65
  2  2 1  68
  2  2 2  68
  3  1 1 106
  3  1 2 104
  3  2 1 102
  3  2 2 101
  4  1 1  98
  4  1 2  98
  4  2 1  95
  4  2 2  98
  5  1 1  40
  5  1 2  45
  5  2 1  40
  5  2 2  44
  6  1 1  87
  6  1 2  88
  6  2 1  90
  6  2 2  88
  7  1 1  98
  7  1 2 105
  7  2 1  89
  7  2 2 104
  8  1 1  98
  8  1 2 101
  8  2 1  99
  8  2 2 100
  9  1 1 122
  9  1 2 121
  9  2 1 118
  9  2 2 118
  10 1 1  75
  10 1 2  76
  10 2 1  71
  10 2 2  71
  11 1 1  80
  11 1 2  83
  11 2 1  82
  11 2 2  80
  12 1 1  94
  12 1 2  95
  12 2 1  98
  12 2 2  95
  13 1 1  98
  13 1 2  98
  13 2 1  98
  13 2 2  94
  14 1 1 105
  14 1 2 111
  14 2 1 111
  14 2 2 107
  15 1 1  93
  15 1 2 105
  15 2 1 101
  15 2 2 104
  ;
run;
options nodate nonumber;
proc print data=rater noobs;
  title 'Original Data';
run;
* Transpose the original data from its
  univariate structure to a multivariate
  structure;
proc transpose data=rater
               out=ratertrans
               (rename=(rater1=rater11
                        rater2=rater12
                        rater3=rater21
                        rater4=rater22)
               drop=_NAME_)
               prefix=rater;
  var count;
  by  slide;
run;
proc print data=ratertrans noobs;
  title 'Original Data Transposed';
run;
data newrater;
  set ratertrans;
* 1. Compute the averages of the readings
     made by each rater;
      x = (rater11 + rater12)/2;
      y = (rater21 + rater22)/2;
* 2. Compute the sum and the difference
     of the averages of the readings;
       u = x + y;
       v = x - y;
run;
* Measure the correlation between u and v;
proc corr data=newrater;
  var u v;
  title;
run;
* Retrieve the value of the mean
  and the standard deviation of v;
proc means data=newrater;
  var v;
  output mean=meanv std=sdv out=reflines;
run;
data lines;
  set reflines;
* 1. Compute the upper and lower limits of v;
       up = meanv + (2*sdv);
       lo = meanv - (2*sdv);
* 2. Convert the variables, up and lo, into
     macro variables;
      call symput('upper',up);
      call symput('lower',lo);
run;
* Produce a scatter plot with the
  upper and lower limits of v;
goptions reset=all;
symbol1 v=plus c=blue;
proc gplot data=newrater;
  plot v*u / vref=&upper &lower lvref=2
             cverf=red;
run;
quit;
```