

A SAS® macro to compare predictive values of diagnostic tests

Lovedeep Gondara, University of Illinois Springfield

ABSTRACT

Medical tests are used for various purposes including diagnosis, prognosis, risk assessment and screening. Statistical methodology is used often to evaluate such types of tests, most frequent measures used for binary data being sensitivity, specificity, positive and negative predictive values. An important goal in diagnostic medicine research is to estimate and compare the accuracies of such tests. In this paper I give a gentle introduction to measures of diagnostic test accuracy and introduce a SAS® macro to calculate generalized score statistic and weighted generalized score statistic for comparison of predictive values using formula's generalized and proposed by Andrzej S. Kosinski.

INTRODUCTION

Diagnostic tests play a key role in medicine to predict or classify presence or absence of a disease or a condition. The performance of a diagnostic test is based on its ability to correctly classify subjects into relevant groups and is often evaluated comparing to a gold standard or existing tests, performance measure becomes even more important when gold standard is either invasive or expensive. In this paper I briefly introduce common measures of diagnostic test accuracy (Sensitivity, Specificity, Positive Predictive Value, Negative Predictive Value) and introduce a macro which can be used to compare predictive values using generalized score statistic and weighted generalized score statistic.

	Disease	Non Disease	Total
Positive	A (True Positive)	B (False Positive)	T _{Test Positive}
Negative	C (False Negative)	D (True Negative)	T _{Test Negative}
	T _{Disease}	T _{Non Disease}	Total

Table 1. Test results vs disease status

SENSITIVITY

Sensitivity of a diagnostic test is defined as the probability that a test will be positive for those with disease.

$$\text{Sensitivity: } \frac{A}{(A+C)} \times 100$$

SPECIFICITY

Specificity of a diagnostic test is defined as the probability that a test will be negative for those without disease.

$$\text{Specificity: } \frac{D}{(D+B)} \times 100$$

Sensitivity and specificity are characteristic of a test, the population has no effect on the results.

PREVALENCE

For a population based study prevalence is defined as proportion of a population having a condition/disease.

$$\text{Prevalence: } \frac{T_{Disease}}{Total} \times 100$$

Positive predictive value (PPV)

PPV is the probability that subjects with a positive test result truly have the disease.

$$PPV: \frac{A}{(A+B)} \times 100$$

Negative predictive value (NPV)

NPV is the probability that subjects with a negative test result truly don't have the disease.

$$NPV: \frac{D}{(D+C)} \times 100$$

PPV and NPV are influenced by the prevalence of disease in the population under study i.e. if tested in a high prevalence setting, it is more likely that a positive tested person will truly have the disease.

COMPARISON OF PREDICTIVE VALUES

GENERALIZED SCORE STATISTIC

Generalized score statistic was presented by Leisenring, Alonzo and Pepe [1]

$$T_{PPV}^{GS} = \frac{\{\sum_{i=1}^{N_p} \sum_{j=1}^{m_i} D_{ij} (Z_{ij} - m_i \bar{Z})\}^2}{\sum_{i=1}^{N_p} \{\sum_{j=1}^{m_i} (D_{ij} - \bar{D})(Z_{ij} - \bar{Z})\}^2}$$

Where $\bar{D} = (\sum_{i=1}^{N_p} \sum_{j=1}^{m_i} D_{ij}) / \sum_{i=1}^{N_p} m_i$, $\bar{Z} = (\sum_{i=1}^{N_p} \sum_{j=1}^{m_i} Z_{ij}) / \sum_{i=1}^{N_p} m_i$, N_p is the number of patients with at least one positive test outcome, and m_i is the number of positive test results for the i^{th} patient.

Under the null hypothesis (H_0), the test statistic is asymptotically χ^2 distributed.

Mathematically equivalent and intuitive reformulation of above equation was presented by Kosinski [2]

$$T_{PPV}^{GS} = \frac{(\widehat{PPV}_1 - \widehat{PPV}_2)^2}{\{\widehat{PPV}_P(1 - \widehat{PPV}_P) + W_{PPV} - 2C_P^{PPV}\} \left(\frac{1}{n_{+++}} + \frac{1}{n_{*++}} \right)}$$

Where

$$\widehat{PPV}_1 = n_{+*D} / n_{+++}$$

$$\widehat{PPV}_2 = n_{**+D} / n_{*++}$$

$$\widehat{PPV}_P = (n_{+*D} + n_{**+D}) / (n_{+++} + n_{*++})$$

$$C_P^{PPV} = (n_{+++} + n_{*++})^{-1} \{n_{++D}(1 - \widehat{PPV}_P)^2 + n_{++\bar{D}} \widehat{PPV}_P^2\}$$

And n_{+*D} = True positive by Test 1 , n_{+**} = Total positive by Test 1, n_{*+D} = True positive by Test 2, n_{**+} = Total positive by Test 2, n_{++D} = True positive from both tests, $n_{++\bar{D}}$ = False positive from both tests

WEIGHTED GENERALIZED SCORE STATISTIC

Kosinski presented weighted generalized score statistic which always reduces to the score statistic when applied to two independent samples.

$$T_{PPV}^{WGS} = \frac{(\widehat{PPV}_1 - \widehat{PPV}_2)^2}{\{\widehat{PPV}_p(1 - \widehat{PPV}_p) - 2C_p^{PPV}\} \left(\frac{1}{n_{+**}} + \frac{1}{n_{**+}} \right)}$$

SAS MACRO PV_SS

In this paper I present a SAS macro `pv_ss` that is capable of performing both above mentioned statistical tests and provide user with value of PPV (Test1), PPV (Test2), NPV (Test1), NPV (Test2), Test Statistic & P-value. Complete macro is provided at the end of this paper.

The macro requires data to be in 0, 1 format with 0 being a negative and 1 positive test result. PROC FREQ is run on input data with sparse statement in order to extract PPV and NPV for both test types.

```
proc freq data=&data noprint;
    table &test1*&dis / outpct out=&test1 sparse ;
    table &test2*&dis / outpct out=&test2 sparse;
    table &test1*&test2*&dis / outpct out=test sparse ;
run;
```

After appending the created datasets together we need to convert output percentages to raw values for calculation of test statistic.

```
data pv;
    set &test1 &test2 test;
    pv=pct_row/100;
run;
```

Then we use PROC SQL to get counts and sums to be used later in calculations.

```
proc sql noprint;
    select pv into :ppv1 from pv where &test1=1 and &&dis=1 and t=1;
    select pv into :ppv2 from pv where &test2=1 and &&dis=1 and t=2;
    .....
    select sum(count) into :d2 from pv where not missing(&test2) and
    &test2=1 and t ne 3;
    select sum(count) into :d1 from pv where not missing(&test1) and
    &test1=1 and t ne 3;
    select sum(count) into :nd2 from pv where not missing(&test2) and
    &test2=0 and t ne 3;
    select sum(count) into :nd1 from pv where not missing(&test1) and
    &test1=0 and t ne 3;
    .....
```

At last a data step is used to compute Test statistic and p-values.

```
data pv_WGS;
    ppv1=&ppv1;
    ppv2=&ppv2;
    npv1=&npv1;
    .....
    cppv=(td*((1-ppv1)**2) + tnd*(ppv1**2))/tt;
    nppv=((ntnd*(npv1**2)) + (ntd*(1-npv1)**2))/ntt;
    .....
    Sp=abs((difppv**2)/((ppv1*(1-ppv1) - 2*cppv)*((1/d1)+(1/d2))));
    pvp=1-probchi(Sp,1);
    .....
```

EXAMPLE

Cass study data was used in this example. It has three test results from Exercise Stress test, Chest pain history and Coronary artery disease, all variables are dichotomized(1=yes, 0=no). We use macro pv_SS to compare the predictive values of Exercise Stress test and Chest pain history to assess Coronary artery disease.

```
filename webdata url
"http://research.fhcrc.org/content/dam/stripe/diagnostic-biomarkers-
statistical-center/files/est1.csv";
```

```

data EST;
  infile webdata dsd firstobs=2;
  input d y1 y2;
  put _infile_;
run;

```

USING GENERALIZED SCORE STATISTIC

```
%pv_SS(data=EST, test1=y1, test2=y2, dis=d, method=gs)
```

Method: Generalized Score Statistic

PPV y1	PPV y2	Test Statistic(PPV)	P- Value(PPV)	NPV y1	NPV y2	Test Statistic(NPV)	P- Value(NPV)
0.87634	0.79819	47.5810	5.2777E-12	0.61122	0.78486	39.5224	3.2431E-10

Table 2. Results of Generalized score statistic

USING WEIGHTED GENERALIZED SCORE STATISTIC

```
%pv_SS(data=EST, test1=y1, test2=y2, dis=d, method=wgs)
```

Method: Weighted Generalized Score Statistic

PPV y1	PPV y2	Test Statistic(PPV)	P- Value(PPV)	NPV y1	NPV y2	Test Statistic(NPV)	P- Value(NPV)
0.87634	0.79819	43.1882	4.972E-11	0.61122	0.78486	34.0577	5.3501E-9

Table 3. Results of Weighted Generalized score statistic

CONCLUSION

Researchers are often faced with a task to compare two test results, so a macro capable of calculating Generalized and Weighted Generalized score statistic for comparison of predictive values has been presented in this paper.

REFERENCES

1. Leisenring W, Alonzo T, Pepe MS. *Comparisons of predictive values of binary medical diagnostic tests for paired designs*. Biometrics. 2000; 56:345–351. DOI: 10.1111/j.0006-341X. 2000.00345.x. [PubMed: 10877288]
2. Kosinski, A.S. (2013). *A weighted generalized score statistic for comparison of predictive values of diagnostic tests*. Stat Med, 32(6):964-77.
3. CASS study data (<http://research.fhcrc.org/diagnostic-biomarkers-center/en/datasets.html>)
4. Christian Stock, Thomas Hielscher (2014). *DTComPair: comparison of binary diagnostic tests in a paired study design*. R package version 1.0.3. URL: <http://CRAN.R-project.org/package=DTComPair>

CONTACT INFORMATION

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

Lovedeep Gondara
Dept. Of Computer Science
University of Illinois Springfield
Lgond2@uis.edu

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.

Macro pv_SS

```
/*  
  
SAS macro pv_SS: Calculates Test Statistic to compare predicted values (pv)  
of a diagnostic test based on generalized score statistic and weighted  
generalized score statistic proposed by Andrzej S.Kosinski (2013)  
  
Author: Lovedeep Gondara  
  
This program is free software; you can redistribute it and/or modify it.  
  
This program is distributed in the hope that it will be useful, but WITHOUT  
ANY WARRANTY.  
  
Please forward any comments to: lgond2@uis.edu  
  
*/  
  
%macro pv_SS(data=, test1=, test2=, dis=, method=);  
  
/*  
  
data --- Name of SAS dataset that should contain information for test1,  
test2 and gold standard result(True result)  
  
test1 --- Name of variable which has information for first test (0=negative,  
1=positive)  
  
test2 --- Name of variable which has information for second test (0=negative,  
1=positive)  
  
dis --- Name of variable with information for true result (0=negative,  
1=positive)  
  
method= Name of method to be used for computation of test statistic  
(gs=generalized score statistic, wgs=weighted generalized score statistic)  
  
*/  
  
/*typical macro call*/  
  
/* %pv_WGS(data=data_set_name, test1=first_test, test2=second_test,  
dis=true_result)*/  
  
  
/*** Using PROC FREQ to get output tables with percentages for both test  
types  
  
so that the PPV and NPV can be extracted ***/  
  
proc freq data=&data noprint;  
    table &test1*&dis / outpct out=&test1 sparse ;  
    table &test2*&dis / outpct out=&test2 sparse;
```

```

        table &test1*&test2*&dis / outpct out=test sparse ;
run;

/** adding extra column to distinguish between both test's */
data &test1;
    set &test1;
    t=1;
run;

data &test2;
    set &test2;
    t=2;
run;

data test;
    set test;
    t=3;
run;

/** Converting percentage to raw values to Test statisitic can be calculated
*** */
data pv;
    set &test1 &test2 test;
    pv=pct_row/100;
run;

/** Getting counts and sums and storing them in macro variables to be used
in calculations */
proc sql noprint;
    select pv into :ppv1 from pv where &test1=1 and &&dis=1 and t=1;
    select pv into :ppv2 from pv where &test2=1 and &&dis=1 and t=2;
    select pv into :npv1 from pv where &test1=0 and &&dis=0 and t=1;
    select pv into :npv2 from pv where &test2=0 and &&dis=0 and t=2;
    select count into :td from pv where &test1=1 and &test2=1 and &&dis=1
and t=3 ;
    select count into :tnd from pv where &test1=1 and &test2=1 and &dis=0
and t=3;

```



```

        select count into :ntd from pv where &test1=0 and &test2=0 and &dis=0
and t=3 ;

        select count into :ntnd from pv where &test1=0 and &test2=0 and &dis=1
and t=3;

        select sum(count) into :di1 from pv where not missing(&test1) and
&dis=1 and &test1=1 and t ne 3;

        select sum(count) into :di2 from pv where not missing(&test2) and
&dis=1 and &test2=1 and t ne 3;

        select sum(count) into :dj1 from pv where not missing(&test1) and
&dis=0 and &test1=0 and t ne 3;

        select sum(count) into :dj2 from pv where not missing(&test2) and
&dis=0 and &test2=0 and t ne 3;

        select sum(count) into :d2 from pv where not missing(&test2) and
&test2=1 and t ne 3;

        select sum(count) into :d1 from pv where not missing(&test1) and
&test1=1 and t ne 3;

        select sum(count) into :nd2 from pv where not missing(&test2) and
&test2=0 and t ne 3;

        select sum(count) into :nd1 from pv where not missing(&test1) and
&test1=0 and t ne 3;

        select sum(count) into :d2i from pv where not missing(&test2) and
&test2=0 and t ne 3;

        select sum(count) into :d1i from pv where not missing(&test1) and
&test1=0 and t ne 3;

        select sum(count) into :d1c from pv where not missing(&test1) and
&test1=1 and &dis=0 and t ne 3;

        select sum(count) into :d2c from pv where not missing(&test2) and
&test2=1 and &dis=0 and t ne 3;

        select sum(count) into :dj1c from pv where not missing(&test1) and
&test1=0 and &dis=1 and t ne 3;

        select sum(count) into :dj2c from pv where not missing(&test2) and
&test2=0 and &dis=1 and t ne 3;

quit;

/** Main calculation's for test statistic and p-values */
data pv_WGS;

    ppv1=&ppv1;
    ppv2=&ppv2;
    npv1=&npv1;
    npv2=&npv2;
    d1=&d1;
    td=&td;

```

```

tnd=&tnd;
ntd=&ntd;
ntnd=&ntnd;
d2=&d2;
nd1=&nd1;
nd2=&nd2;
d1i=&d1i;
d2i=&d2i;
di=&d1i+&d2i;
dj=&dj1+&dj2;
tt=&d1+&d2;
ntt=&nd1+&nd2;
nd=&d1c+&d2c;
nnd=&dj1c+&dj2c;
ppvp=di/(&d1+&d2);
npvp=dj/(&d1i+&d2i);
difppv=abs(ppv1-ppv2);
difnpv=abs(npv1-npv2);
cppv=(td*(1-ppv)**2) + tnd*(ppv**2)/tt;
nppv=(ntnd*(npvp**2) + (ntd*(1-npv)**2))/ntt;
%if &method=wgs %then %do;
    Sp=abs((difppv**2)/((ppv*(1-ppv) - 2*cppv)*(1/d1)+(1/d2)));
    Sn=abs((difnpv**2)/((npvp*(1-npv) - 2*nppv)*(1/nd1)+(1/nd2)));
    pvp=1-probchi(Sp,1);
    pvn=1-probchi(Sn,1);
%end;
%if &method=gs %then %do;
    wppv=((2*ppv)-ppv1-ppv2)*((2*ppv)-1);
    wnpv=((2*npvp)-npv1-npv2)*((2*npvp)-1);
    Sp=abs((difppv**2)/((ppv*(1-ppv)+wppv - 2*cppv)*(1/d1)+(1/d2)));
    Sn=abs((difnpv**2)/((npvp*(1-npv)+wnpv - 2*nppv)*(1/nd1)+(1/nd2)));
    pvp=1-probchi(Sp,1);
    pvn=1-probchi(Sn,1);
%end;
run;

```

```

%if &method=gs %then %let title=Method: Generalized Score Statistic;
%if &method=wgs %then %let title=Method: Weighted Generalized Score
Statistic;

/**** Printing results ****/
proc print data=pv_WGS noobs label;
    var ppv1 ppv2 Sp pvp npv1 npv2 Sn pvn;
    label ppv1="PPV &test1" ppv2="PPV &test2" npv1="NPV &test1" npv2="NPV
&test2" Sp='Test Statistic(PPV) ' pvp='P-Value(PPV) ' Sn='Test Statistic(NPV) '
pvn='P-Value(NPV) ';
    title1 "&title";
run;

/**** Garbage collection ****/
proc delete data=&test1 &test2 test pv pv_WGS;
run;

%mend pv_SS;

```