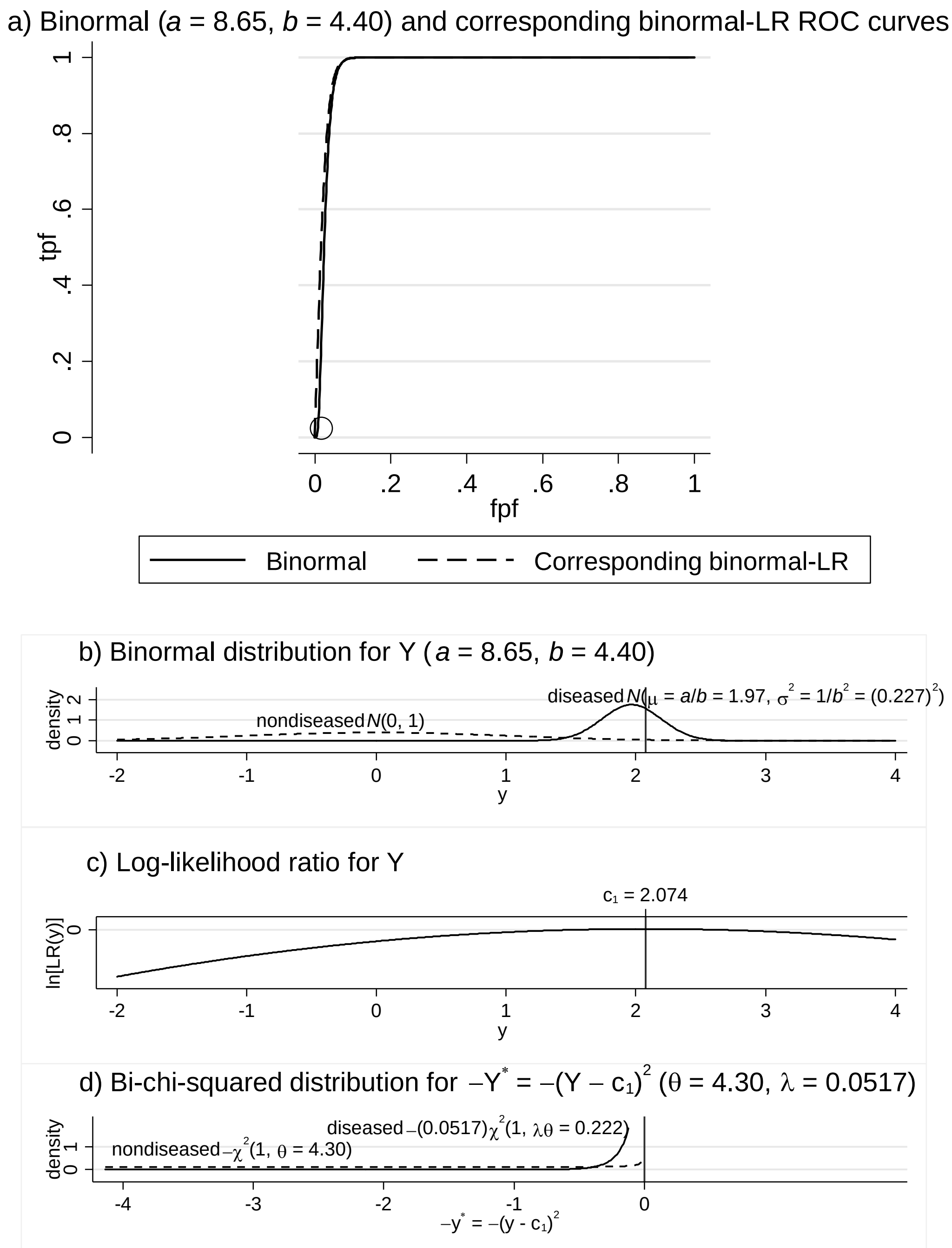
Web-based Supporting Materials for “Equivalence of binormal likelihood-ratio and bi-chi-squared ROC curve models” by Stephen L. Hillis

Table of Contents

Figure S1: Binormal ROC curve (with $b > 1$) and related plots

The binormal ROC curve (solid line) in (a) has an almost indiscernable hook in the lower left corner, as indicated by the circle. For the corresponding binormal random variable Y, the binormal conditional densities and the log-likelihood ratio are displayed in (b) and (c), respectively, where $c_1 = -ab/(1-b^2)$ is the threshold where the log-likelihood ratio attains its maximum. The bi-chi-squared conditional densities for $-Y^* = -(Y - c_1)^2$ are displayed in (d), and the binormal-LR ROC curve, based on $-Y^*$, is the dashed line in (a). Notes: the nondiseased and diseased densities are denoted by the dashed and solid lines, respectively, in (b) and (d); $\chi^2(1, \nu)$ denotes a chi-squared distribution with one degree of freedom and noncentrality parameter ν; LR: likelihood ratio function; $\lambda = 1/b^2$; $\theta = a^2 b^2 / (1-b^2)^2$.

(NOTE: figure is on next page)


a) Binormal (a = 8.65, b = 4.40) and corresponding binormal-LR ROC curves
tpf
0
.2
.4
.6
.8
1
fpf
Binormal
Corresponding binormal-LR
b) Binormal distribution for Y (a = 8.65, b = 4.40)
density
diseased N(μ = a/b = 1.97, σ² = 1/b² = (0.227)²)
nondiseased N(0, 1)
-2
-1
0
1
2
3
4
y
c) Log-likelihood ratio for Y
c₁ = 2.074
ln[LR(y)]
d) Bi-chi-squared distribution for −Y* = −(Y − c₁)² (θ = 4.30, λ = 0.0517)
diseased −(0.0517)χ²(1, λθ = 0.222)
nondiseased −χ²(1, θ = 4.30)
-4
-3
-2
-1
0
−y* = −(y - c₁)²

Appendix S1: Simple bi-chi-squared program without macros

```
*******************************************************************
This program is for the following paper: "Eqivalence of binormal
     likelihood-ratio and bi-chi-squared ROC curve models" by
     Stephen L. Hillis.
*******************************************************************
This program performs ML estimation for the bi-chi-square model and produces
the output for reader 1 using modality 1 in Table 4 for the Van Dyke et al data. This
program is written without macros to make it easier to understand.

Both of the AUC formulas in Table 3 are used and yield the same result (as they
should).  The AUC formula from Table 3(d) is labeled "Hillis_AUC" and the formula
from Table 3(e) is labeled "Metz_AUC."
;
****VanDyke data -- these data can be downloaded from perception.radiology.uiowa.edu.
The data set name is VanDyke_anova.sas7bdat.  Click on
perception.radiology.uiowa.edu --> Software --> Receiver Operating Characteristic (ROC)
--> SAS Programs to Perform MRMC Analysis.  The file
can be found in "Installation Files for SAS MRMC Programs."
;
libname source "D:\Dorfman\data sets -- important";
  **NOTE: specify above the directory where VanDyke data is stored**;

proc nlmixed data=source.VanDyke_anova gconv = 0 maxit = 500 ;
  where reader = 1 & treatment = 1;
  parms  delta1 .681 delta2 = .13  delta3 = .145 delta4 = .029
    log_lambda =  1.24 log_theta =  0.50 ; **deltas are the initial 4
rating-level nondiseased probabilities, e.g., delta1 = pr(R = 1|nondiseaed);
  bounds  delta1-delta4 > 0;
  theta = exp(log_theta);
  lambda = exp(log_lambda);
  array delta[5] delta1 - delta5;
  array spec[5] spec1 - spec5;
  array fpf[5] fpf1 - fpf5;
  array tpf[5] tpf1 - tpf5;
  spec[1] = 0; **specificity = 0 for rating = 1**;
  temp = 0;
  do i = 2 to 5; **compute other specificities**;
    spec[i] = min(1,temp + delta[i-1]);
temp = spec[i];
  end;
  delta[5] = max(0, 1 - spec[5]);
  do i = 1 to 5;
    fpf[i] = 1 - spec[i];
  end;
  tpf[1] = 1;
  do i = 2 to 5;
    if fpf[i] = 0 or fpf[i] = 1 then do;
  if fpf[i] = 0 then tpf[i] = 0;
  if fpf[i] = 1 then tpf[i] = 1;
end;
    else do;
      if lambda >= 1 then tpf[i] = 1 - probchi(lambda**-1 * cinv(1 -
fpf[i],1,theta),1,lambda*theta);
  if lambda < 1 then tpf[i] = probchi(lambda**-1 * cinv(fpf[i],1,theta),1,lambda*theta);
end;
  end;
  if rating=5 then z= (truth = 0)*fpf[5] + (truth=1)*tpf[5];
  do i = 1 to 4;
    if rating = i then z= (truth = 0)*(fpf[i] -fpf[i+1]) + (truth=1)*(tpf[i] -tpf[i+1]);
  end;
  if z>1e-200 then ll = log(z); **to avoid undefined contribution**;
```

```
    else ll= log(1e-200);
  model rating  ~ general(ll);
  u1 = (lambda*sqrt(theta) - sqrt(theta))/sqrt(1+lambda);
  u2 = (lambda*sqrt(theta) + sqrt(theta))/sqrt(1+lambda);
  rho = (-1 + lambda)/(1 + lambda);
  Metz_arg = sqrt(theta)*(-1+lambda)/sqrt(1+lambda); **this is d_a/sqrt(2)**;
  *The next 3 statements avoid undefined values for the probbnrm and probnorm functions*;
  if u1 < -4 then u1 = -4; if u1 > 4 then u1 = 4;
  if u2 < -4 then u2 = -4; if u2 > 4 then u2 = 4;
  If Metz_arg <-4 then metz_arg = -4; if metz_arg > 4 then metz_arg = 4;
  **optional estimate statements below**;
    *Note: AUC_Hillis and AUC_Metz use formulas (d) and (e) in Table 3, respectively*;
  estimate "AUC_Hillis" (lambda>1)*(probbnrm(u1,u2,rho) + probbnrm(-u1,-u2,rho))
      + (lambda<1)*(1 - probbnrm(u1,u2,rho) - probbnrm(-u1,-u2,rho));
  estimate "AUC_Metz" probnorm(metz_arg)
      + 2*probbnrm(-metz_arg,0,-2*sqrt(lambda)/(lambda+1));
  estimate "lambda" lambda;
  estimate "theta" theta;
  estimate "a" sqrt((lambda - 1)**2 * theta) /sqrt(lambda);
  estimate "b" 1/sqrt(lambda);
run;
```

Appendix S2: Bi-chi-squared program with macros

```
*Bi-chi-squared program with macros for paper 10-05-14 revised 07-04-15b with Hillis and
MetzAUC.sas*;
*Appendix S2: Bi-chi-squared program with macros
*******************************************************************
This program is for the following paper: "The equivalence of the binormal
     likelihood-ratio and bi-chi-squared ROC curve models," by
     Stephen L. Hillis.
*******************************************************************
This program performs ML estimation for the bi-chi-square model and produces
the output for Table 4 in the paper. Both of the AUC formulas in Table 3 are
used and yield the same result (as they should).  The AUC formula from Table 3(d)
is labeled "Hillis_AUC" and the formula from Table 3(e) is labeled "Metz_AUC."

Output from the program is included.

The only statement that needs to be changed to run the program is the libname statment
below -- the appropriate directory for the VanDyke data must be specified.
;

****VanDyke data -- these data can be downloaded from perception.radiology.uiowa.edu.
The data set name is VanDyke_anova.sas7bdat.  Click on
perception.radiology.uiowa.edu --> Software --> Receiver Operating Characteristic (ROC)
--> SAS Programs to Perform MRMC Analysis.  The file
can be found in "Installation Files for SAS MRMC Programs."
;

libname source "D:\Dorfman\data sets -- important"; *"D:\Dorfman\Obuchowski\Data sets";
  **IMPORTANT NOTE: specify above the directory where VanDyke data is stored**;

%macro BYCHI;
proc nlmixed data=&dataset   gconv = 0 maxit = 500 ; where &where;
  ods output additionalEstimates = parmsout parameterEstimates = parm_estimates;
  %let numcat_minus_1 =%eval(&numcat - 1);
  parms  delta1 =&delta1 delta2 = &delta2  delta3 = &delta3 delta4 = &delta4
    log_lambda =  &log_lambda log_theta =  &log_theta ; **deltas are the initial p-1
rating level nondiseased probabilities, e.g., delta1 = pr(R = 1|nondiseaed);
  bounds  delta1-delta&numcat_minus_1 > 0;
  theta = exp(log_theta);
  lambda = exp(log_lambda);
  array delta[&numcat] delta1 - delta&numcat;
  array spec[&numcat] spec1 - spec&numcat;
  array fpf[&numcat] fpf1 - fpf&numcat;
  array tpf[&numcat] tpf1 - tpf&numcat;
  spec[1] = 0; **specificity = 0 for rating = 1**;
  temp = 0;
  do i = 2 to &numcat; **compute other specificities**;
    spec[i] = min(1,temp + delta[i-1]);
temp = spec[i];
  end;
  delta[&numcat] = max(0, 1 - spec[&numcat]);
  do i = 1 to &numcat;
    fpf[i] = 1 - spec[i];
  end;
  tpf[1] = 1;
  do i = 2 to &numcat;
    if fpf[i] = 0 or fpf[i] = 1 then do;
  if fpf[i] = 0 then tpf[i] = 0;
  if fpf[i] = 1 then tpf[i] = 1;
end;
    else do;
```

```
      if lambda >= 1 then tpf[i] = 1 - probchi(lambda**-1 * cinv(1 -
fpf[i],1,theta),1,lambda*theta);
  if lambda < 1 then tpf[i] = probchi(lambda**-1 * cinv(fpf[i],1,theta),1,lambda*theta);
end;
  end;
  if rating=&numcat then z= (truth = 0)*fpf[&numcat] + (truth=1)*tpf[&numcat];
  do i = 1 to &numcat_minus_1;
    if rating = i then z= (truth = 0)*(fpf[i] -fpf[i+1]) + (truth=1)*(tpf[i] -tpf[i+1]);
  end;
  if z>1e-200 then ll = log(z); **to avoid undefined contribution**;
     else ll= log(1e-200);
  model rating  ~ general(ll);
  u1 = (lambda*sqrt(theta) - sqrt(theta))/sqrt(1+lambda);
  u2 = (lambda*sqrt(theta) + sqrt(theta))/sqrt(1+lambda);
  y =sqrt(theta);
  rho = (-1 + lambda)/(1 + lambda);
  Metz_arg = sqrt(theta)*(-1+lambda)/sqrt(1+lambda); **this is d_a/sqrt(2)**;
  *The next 3 statements avoid undefined values for the probbnrm and probnorm functions*;
  if u1 < -4 then u1 = -4; if u1 > 4 then u1 = 4;
  if u2 < -4 then u2 = -4; if u2 > 4 then u2 = 4;
  If Metz_arg <-4 then metz_arg = -4; if metz_arg > 4 then metz_arg = 4;
 **optional estimate statements below**;
    *Note: AUC_Hillis and AUC_Metz use formulas (d) and (e) in Table 3, respectively.*;
  estimate "AUC_Hillis" (lambda>1)*(probbnrm(u1,u2,rho) + probbnrm(-u1,-u2,rho))
      + (lambda<1)*(1 - probbnrm(u1,u2,rho) - probbnrm(-u1,-u2,rho));
  estimate "AUC_Metz" probnorm(metz_arg)
      + 2*probbnrm(-metz_arg,0,-2*sqrt(lambda)/(lambda+1));
  estimate "lambda" lambda;
  estimate "theta" theta;
  estimate "a" sqrt((lambda - 1)**2 * theta) /sqrt(lambda);
  estimate "b" 1/sqrt(lambda);

run;
data parmsout&reader.&treatment; set parmsout; reader = &reader; treatment = &treatment; run;
%mend BYCHI;

data VanDyke; set source.VanDyke_anova; run;

****This next part is only for determining the empirical fpf values for ratings 1-4 that will be
    used as the initial delta values in the program*************************;

ods listing close;  **this statement suppresses the printing of output
  not needed for the bi-chi-square program**;
proc freq data = vandyke ;
ods output crosstabfreqs = freqtable;
  table treatment*reader*truth*rating; run;
proc print data = freqtable; run;
data freqtable1; set freqtable; if truth = 0; if rating ne .;
  keep treatment reader rating rowpercent;
data freqtable2; set freqtable1;
  delta = rowpercent/100;
  if rating ne 5;
ods listing;  **now printing turned back on**;
proc print; format delta 5.3;
  title "Starting values for delta";
  title3 "Zero values will be changed to .001";
  title4 "and totals constrained  to not exceed 1";
  title5 "(by substracting a little from large deltas)";
  TITLE6 "as indicated in bi-chi-square program macros statements below";
run;
title "";

/* below are results of above print statement
```

Starting values for delta

Zero values will be changed to .001
and totals constrained to not exceed 1
(by substracting a little from large deltas)
as indicated in bi-chi-square program macro statements below

Obs	treatment	reader	rating	Row Percent	delta
1	1	1	1	68.1159	0.681
2	1	1	2	13.0435	0.130
3	1	1	3	14.4928	0.145
4	1	1	4	2.8986	0.029
5	1	2	1	0.0000	0.000
6	1	2	2	86.9565	0.870
7	1	2	3	8.6957	0.087
8	1	2	4	2.8986	0.029
9	1	3	1	30.4348	0.304
10	1	3	2	50.7246	0.507
11	1	3	3	7.2464	0.072
12	1	3	4	8.6957	0.087
13	1	4	1	89.8551	0.899
14	1	4	2	4.3478	0.043
15	1	4	3	5.7971	0.058
16	1	4	4	0.0000	0.000
17	1	5	1	56.5217	0.565
18	1	5	2	27.5362	0.275
19	1	5	3	13.0435	0.130
20	1	5	4	1.4493	0.014
21	2	1	1	33.3333	0.333
22	2	1	2	34.7826	0.348
23	2	1	3	21.7391	0.217
24	2	1	4	10.1449	0.101
25	2	2	1	8.6957	0.087
26	2	2	2	81.1594	0.812
27	2	2	3	10.1449	0.101
28	2	2	4	0.0000	0.000
29	2	3	1	36.2319	0.362
30	2	3	2	44.9275	0.449
31	2	3	3	11.5942	0.116
32	2	3	4	5.7971	0.058
33	2	4	1	63.7681	0.638
34	2	4	2	30.4348	0.304
35	2	4	3	5.7971	0.058
36	2	4	4	0.0000	0.000
37	2	5	1	30.4348	0.304
38	2	5	2	56.5217	0.565
39	2	5	3	13.0435	0.130
40	2	5	4	0.0000	0.000

```
******end of empirical fpf value determination section******;
*/

****************starting estimates from binormal fit*******************
  Below are the estimated binormal a and b values and AUC for the VanDyke data and the
corresponding lambda, theta, log(lambda) and log(theta) values.  The
log(lambda) and log(theta) values are used as the initial starting values
in the bi-chi-square program.
  These binormal estimates were obtained using the software OR/DBM MRMC for SAS 3.0
which is avaiilable for download from perception.radiology.uiowa.edu.
*****************************************************************;
/*
```

```
                                                                                log_        log_
Obs   treatment   reader     AUC        a         b       lambda    theta     lambda      theta

  1       1          1     0.93316    1.7022    0.5368    3.4707    1.6473    1.24435    0.49917
  2       1          2     0.88957    1.4037    0.5610    3.1778    1.3202    1.15618    0.27782
  3       1          3     0.92920    1.7409    0.6346    2.4830    3.4216    0.90945    1.23012
  4       1          4     0.97046    1.9255    0.2015   24.6279    0.1636    3.20388   -1.81060
  5       1          5     0.83259    1.0630    0.4635    4.6545    0.3938    1.53784   -0.93188
  6       2          1     0.95082    1.8502    0.5030    3.9525    1.5520    1.37436    0.43956
  7       2          2     0.93460    1.6552    0.4473    4.9976    0.8568    1.60897   -0.15461
  8       2          3     0.92755    1.6220    0.4878    4.2017    1.0783    1.43549    0.07542
  9       2          4     1.00000    7.1233    0.8806    1.2896  780.1023    0.25435    6.65943
 10       2          5     0.94481    1.7329    0.4221    5.6124    0.7922    1.72499   -0.23298
*/

*****************************************************************
  Below are the statements for calling the bi-chi-square algorithm macro to obtain the
bi-chi-square results given in the paper.
*****************************************************************;
%let numcat = 5;

%let dataset = vandyke;
%let log_lambda = 1.24; %let log_theta = 0.50;
%let reader = 1; %let treatment = 1;
%let where =  reader = &reader & treatment = &treatment;
%let delta1 = .681;
%let delta2 = .13;
%let delta3 = .145;
%let delta4 = .029;
%BYCHI;

%let log_lambda = 1.16; %let log_theta = 0.28;
%let reader = 2; %let treatment = 1;
%let where =  reader = &reader & treatment = &treatment;
%let delta1 = .001;
%let delta2 = .87;
%let delta3 = .087;
%let delta4 = .029;
%BYCHI;

%let log_lambda = .91; %let log_theta = 1.23;
%let reader = 3; %let treatment = 1;
%let where =  reader = &reader & treatment = &treatment;
%let delta1 = .304;
%let delta2 = .507;
%let delta3 = .072;
%let delta4 = .087;
%BYCHI;

%let log_lambda = 3.20; %let log_theta = -1.81;
%let reader = 4; %let treatment = 1;
%let where =  reader = &reader & treatment = &treatment;
%let delta1 = .89;  *changed from .899*;
%let delta2 = .043;
%let delta3 = .058;
%let delta4 = .001; *changed from 0*;
%BYCHI;

%let log_lambda = 1.54; %let log_theta = -0.93;
%let reader = 5; %let treatment = 1;
%let where =  reader = &reader & treatment = &treatment;
%let delta1 = .565;
%let delta2 = .275;
```

```
%let delta3 = .13;
%let delta4 = .014;
%BYCHI;

%let log_lambda = 1.37; %let log_theta = 0.44;
%let reader = 1; %let treatment = 2;
%let where =  reader = &reader & treatment = &treatment;
%let delta1 = .333;
%let delta2 = .348;
%let delta3 = .217;
%let delta4 = .101;
%BYCHI;

%let log_lambda = 1.61; %let log_theta = -0.15;
%let reader = 2; %let treatment = 2;
%let where =  reader = &reader & treatment = &treatment;
%let delta1 = .087;
%let delta2 = .812;
%let delta3 = .099; *changed from 0.101;
%let delta4 = .001; *changed from 0.000*;
%BYCHI;

%let log_lambda = 1.44; %let log_theta = 0.08;
%let reader = 3; %let treatment = 2;
%let where =  reader = &reader & treatment = &treatment;
%let delta1 = .362;
%let delta2 = .449;
%let delta3 = .116;
%let delta4 = .058;
%BYCHI;

%let log_lambda = 0.25; %let log_theta = 6.66;
%let reader = 4; %let treatment = 2;
%let where =  reader = &reader & treatment = &treatment;
%let delta1 = .638;
%let delta2 = .304;
%let delta3 = .055; *changed from 0.58*;
%let delta4 = .001; *changed from 0.000*;
%BYCHI;

%let log_lambda = 1.72; %let log_theta = -0.23;
%let reader = 5; %let treatment = 2;
%let where =  reader = &reader & treatment = &treatment;
%let delta1 = .304;
%let delta2 = .565;
%let delta3 = .128; *changed from 0.130*;
%let delta4 = .001; *changed from 0.000*;
%BYCHI;

data parmsout_all; set parmsout11 parmsout21 parmsout31 parmsout41 parmsout51
  parmsout12 parmsout22 parmsout32 parmsout42 parmsout52;
  keep label estimate reader treatment; run;
proc transpose data = parmsout_all out = parmsout_all_tr; id label; by treatment reader;
   run;

options ls = 80;
proc print data = parmsout_all_tr; format AUC_Hillis AUC_Metz a b lambda theta 15.6;
  id treatment reader; var  auc_hillis auc_metz lambda theta a b ;run;

*************************************************
Below are the results from the above print statment, which include the Bi-chi-square
AUC, lambda, and theta values for Table 4.
```

```
treatment reader      AUC_Hillis        AUC_Metz          lambda           theta
    1        1          0.934040        0.934040        3.418921        1.706011
    1        2          0.891071        0.891071        3.172871        1.324855
    1        3          0.928886        0.928886        2.532211        3.239212
    1        4          0.977460        0.977460      786.713272        0.000017
    1        5          0.840559        0.840559        9.366031        0.059426
    2        1          0.951936        0.951936        3.788976        1.697362
    2        2          0.925992        0.925992       73.215915        0.000022
    2        3          0.930432        0.930432        3.940212        1.234458
    2        4          0.999937        0.999968        1.283937      780.544368
    2        5          0.942687        0.942687       12.075803        0.217395

treatment reader               a                 b
    1        1          1.708709          0.540823
    1        2          1.404080          0.561402
    1        3          1.732959          0.628420
    1        4          0.114029          0.035653
    1        5          0.666394          0.326755
    2        1          1.866685          0.513735
    2        2          0.039241          0.116868
    2        3          1.645724          0.503779
    2        4          7.000815          0.882527
    2        5          1.486081          0.287768

***************************************************************************
  Below are results when I use different starting values for reader 3 and modality 1, which gives me
the same answer as PROPROC;

%let numcat = 5;
%let dataset = vandyke;
%let log_lambda =   -4 -2   0  2  4  ;
%let log_theta =    -4 -2   0 2 4 ;
%let reader = 3; %let treatment = 1;
%let where =  reader = &reader & treatment = &treatment;
%let delta1 = .304;
%let delta2 = .507;
%let delta3 = .072;
%let delta4 = .087;
%BYCHI;

data parmsout31; set parmsout31; keep label estimate reader treatment; run;
proc transpose data = parmsout31 out = parmsout31_tr; id label; by treatment reader;
   run;
proc print data = parmsout31_tr; format AUC_hillis AUC_Metz a b lambda theta 15.6;
  id treatment; var  auc_hillis auc_metz lambda theta a b ;run;

**Results from above print statement:
      treatment         AUC_Hillis          AUC_Metz            lambda
          1               0.907833           0.907833         46.925054

      treatment              theta                 a                 b
          1               0.000056           0.049949          0.145981

;
```