/* Validation model including calibration and discrimination
DECEMBER 2007

Programmer: Adrien FRANCAIS
Statistical Enginner

Albert Bonniot Institute
Team 11-Outcome of cancer and critical illnesses
UJF-INSERM U823
38706 LA TRONCHE CEDEX
FRANCE
04 76 54 94 00

The main MACRO : %VALIDATION
Macro to test if the model is correct for the calibration and discrimination

TABLEENTREE : work table which includes probability and outcome variable.
VARINTERET: binary outcome variable
PROBA : name of the outcome probability precedently obtained
SORTIE : place where the Hosmer Lemeshow table output in Excel is located

1- MACRO included in the main program : %CALIBRATION
Macro to test the calibration of the model. We obtain the Khi2 of Hosmer-Lemeshow and if it is inferior to 15.5,
the model is well calibrated.

2- MACRO included in the main program : %GRAPHCALIBRATION
Macro to realize the graph of calibration

3- MACRO included in the main program : %DISCRIMINATION
Macro to test the discrimination of the model

4- MACRO included in the main program : %ROC
Macro to realize the graph of the ROC curve

5- MACRO included in the main program : %AUC
Macro to obtain the Area Under Curve. If it is superior to 0.80 then the model discriminates correctly population.

6- MACRO included in the main program : %SUMMARY
Macro which summarize results about discrimination and calibration

Before using this macro, you have to :

- Prepare your database with correct variables
- Compile the macro
- Execute the macro :

```
%validation(tableentree=sepsisdebut,varinteret=dc7jours,proba=pgenmodvalid,
sortie="C:\Documents and Settings\Adrien\Bureau\resutout.xls");
```

You can see LOG screen for information of algorithm.

If you use this macro for work to be published please use the following citation:

Francais Adrien
Validation model including calibration and discrimination

```
*/

/*MAIN MACRO*/
%macro validation(tableentree=,varinteret=,proba=,sortie=);
%let error=0;
%let errormodal=0;
/*some errors messages*/
%if &tableentree =  %then %do;
	%put "ERROR: You didn't give the work table : Please enter the name of work
table";%let error=1;
%end;
%if &varinteret =  %then %do;
	%put "ERROR: You didn't give outcome variable : Please enter a binary
variable";%let error=1;
%end;
%if &proba =  %then %do;
	%put "ERROR: You didn't give the name of probability variable : Please enter a
quantitative variable";%let error=1;
%end;
%if &sortie =  %then %do;
	%put "ERROR: You didn't give the name of final table : Please enter the name of the
final table";%let error=1;
%end;

/*to test if the outcome variable is binary*/
%if &tableentree ne and &error=0 %then %do;
%put "We test if the outcome variable is binary";
	ods output onewayfreqs=aaa(keep=&varinteret);
	title "Test of variable &varinteret";
	proc freq data=&tableentree;tables &varinteret;run;
	data aaaa;set aaa;id=1;run;
	data aaaaa;set aaaa;by id;
	retain errormodal 0;
	if first.id and &varinteret ne 0 then errormodal=1;
	if last.id and &varinteret ne 1 then errormodal=1;
	call symput ('errormodal',errormodal);
	run;
%end;
%if &errormodal = 1 %then %do;
```

```
	%put "ERROR: You didn't give a correct outcome variable : Please enter a binary
variable";%let error=1;
%end;


%if &error = 1 %then %do;
	%put "PLEASE CORRECT ERROR(S) DETAILED ABOVE";
%end;

/*if there is no error then we start the macro*/
%if &error ne 1 %then %do;
	%calibration (&tableentree,&proba,&varinteret,&sortie);
	%graphcalibration;
	%discrimination(&tableentree,&proba,&varinteret);
	%roc;
	%auc;
	%summary;
%end;
%mend;



/* MACRO TO CALCULATE THE CHI2 OF HOSMER LEMESHOW*/
%macro calibration (entree,prob,varinteret,racine);
/*we recover size of each group*/
data dsfrfr;set &entree;retain k l 0;k=k+1;l=1;run;
data dsfdsf;set dsfrfr;by l;if last.l;inter=round(k/10);call symput ('seuil',inter);keep k inter;run;

/*we sort data by probability*/
proc sort data=&entree;by &prob;run;
data dsds;set &entree;probjmoins1=lag(&prob);keep obs &prob &varinteret probjmoins1;run;

/*we create ten groups*/
data logit2;set dsds;retain compt 0 somprob 0 group 1 outcome 0;compt=compt+1;
if compt >=&seuil  and compt<&seuil*2 and group=1 and &prob ne probjmoins1 then
do;group=group+1;outcome=0;somprob=0;end;
if compt >=&seuil*2 and compt<&seuil*3 and group=2 and &prob ne probjmoins1 then
do;group=group+1;outcome=0;somprob=0;end;
if compt >=&seuil*3  and compt<&seuil*4 and group=3 and &prob ne probjmoins1 then
do;group=group+1;outcome=0;somprob=0;end;
if compt >=&seuil*4  and compt<&seuil*5 and group=4 and &prob ne probjmoins1 then
do;group=group+1;outcome=0;somprob=0;end;
if compt >=&seuil*5  and compt<&seuil*6 and group=5 and &prob ne probjmoins1 then
do;group=group+1;outcome=0;somprob=0;end;
if compt >=&seuil*6  and compt<&seuil*7 and group=6 and &prob ne probjmoins1 then
do;group=group+1;outcome=0;somprob=0;end;
if compt >=&seuil*7  and compt<&seuil*8 and group=7 and &prob ne probjmoins1 then
do;group=group+1;outcome=0;somprob=0;end;
if compt >=&seuil*8  and compt<&seuil*9 and group=8 and &prob ne probjmoins1 then
do;group=group+1;outcome=0;somprob=0;end;
```

```
if compt >=&seuil*9  and group=9 and &prob ne probjmoins1 then
do;group=group+1;outcome=0;somprob=0;end;
outcome=outcome+&varinteret;somprob=somprob+&prob;run;

data ddsq;set logit2;retain var1 var2 0;
if group in(1 3 5 7 9)then var1=compt;
if group in(2 4 6 8 10)then var2=compt;
if group in(2 4 6 8 10) then eff=compt-var1;
if group in(1 3 5 7 9) then eff=compt-var2;
drop var1 var2 compt;rename eff=compt;run;

data partition;set ddsq;by group;
retain pas_outcome_observe 0 pas_outcome_theorique 0 effectif 0 eff 0;
rename somprob=nb_outcome_attendu outcome=nb_outcome_observe compt=effectif;
pas_outcome_observe=compt-outcome;pas_outcome_theorique=compt-somprob;
if last.group then output;
keep  group somprob outcome compt effectif pas_outcome_observe pas_outcome_theorique
;run;

/*part to obtain the maximum value of future graph*/
data uyuy;set partition;rapppred=nb_outcome_attendu/&effoutcome;
if group=10 then output;maxi=round(rapppred+0.07,.01);call symput('max',maxi);run;

/*Hosmer -Lemeshow test*/
data tablesortiehosmer;set partition;retain chi2hl 0 chi2bis 0 somchi2bis 0 somchi2hl 0 somtot
0 x 0 y 0;
chi2hl=((pas_outcome_observe-pas_outcome_theorique)*(pas_outcome_observe-
pas_outcome_theorique))/pas_outcome_theorique;
/*last value give us the Hosmer Lemeshow Khi2*/
chi2bis=((nb_outcome_observe-nb_outcome_attendu)*(nb_outcome_observe-
nb_outcome_attendu))/nb_outcome_attendu;
somchi2bis=somchi2bis+chi2bis;somchi2hl=somchi2hl+chi2hl;
somtot=somchi2hl +somchi2bis;if group=10 then do;x=&max;y=&max;end;
if group=10 then call symput('chi2hl',somtot);run;

data fesd;
put group 12. effectif 12. nb_outcome_observe 12. nb_outcome_attendu 12.1
pas_outcome_observe 12. pas_outcome_theorique 12.1 somtot 12.1 x 12.3 y 12.3;
run;

data tablesortiehosmer;merge fesd tablesortiehosmer (keep=group  effectif
nb_outcome_observe
nb_outcome_attendu  pas_outcome_observe  pas_outcome_theorique  somtot x y);run;

/*we export the final table to Excel*/
proc export data =tablesortiehosmer outfile=&racine dbms=excel replace;run;

ods output onewayfreqs=one(keep=frequency &varinteret);proc freq data=sepsisdebut;tables
&varinteret;run;ods trace off;
```

```
data one2;set one;if &varinteret=1 then do;call symput('effoutcome',frequency);end;run;

/*we calculate ratios for the graph*/
data tablesortiehosmer;set tablesortiehosmer;
rapppred=nb_outcome_attendu/&effoutcome;
rappobs=nb_outcome_observe/&effoutcome;
run;

%put "HOSMER LEMESHOW KHI2 WAS &chi2hl";
%mend;


/*macro to draw the graph of calibration*/
%macro graphcalibration;

/*we recover the maximal value of the graphic*/
data dsfsdf;set tablesortiehosmer;
if group=10 then output;maxi=round(rapppred+0.10,.01);
call symput('maxgraph',maxi);run;

title 'Ratios of outcome predicted / observed';
filename gsf pipe 'lp -d printername';
goptions reset=all rotate=landscape
      gsfmode=replace gaccess=sasgastd gsfname=gsf cback=white;
**************************************************************************;
title1 height=2.0 font=swiss 'Figure 1';
title2 height=1.5 font=swiss 'Ratios of outcome predicted / observed' ;
symbol1 color=black font=swissx interpol=join line=1;
symbol2 color=black font=swissx interpol=join line=2;
axis1 label=(height=1.0 font=swiss 'percentage_outcome_observed')
      value=(height=1 font=swiss ) order=(0 to &maxgraph by 0.05)  minor=none
      length=15 cm offset=(1,1)
      ;
axis2 label=(height=1.0 font=swiss 'percentage_outcome_predicted' rotate=90)
      value=(height=1 font=swiss) length=15 cm
      minor=none order=(0 to &maxgraph by 0.05) offset=(0,0);

proc gplot data=tablesortiehosmer;
plot rapppred*rappobs x*y=1 /overlay haxis=axis1 vaxis=axis2;
footnote ' ' h=3;run;
%mend;


/*Macro to test the discrimination*/
%macro discrimination(table,prob,varinteret);

/*we recover the effectif for each value of outcome variable*/
ods output onewayfreqs=one(keep=frequency &varinteret);
proc freq data=sepsisdebut;tables &varinteret;run;
```

```
ods trace off;

data one2;set one;
if &varinteret=1 then do;call symput('effoutcome',frequency);end;
if &varinteret=0 then do;call symput('effpasoutcome',frequency);end;
run;

/*threshold every 2%*/
data classif;set &table;retain i 0;
do i=0 to 1 by 0.02;
     if &prob>=i and &prob<i+0.02 then problevel=i+0.02;
end;run;


data classif2;set classif;by problevel;
retain outcome 0 correctevent &effoutcome correctnonevent 0 nonevent 0 incorrectevent
&effpasoutcome incorrectnonevent 0 ;
correctevent=&effoutcome-outcome;
correctnonevent=nonevent;
incorrectevent=&effpasoutcome-correctnonevent;
incorrectnonevent=outcome;
outcome=outcome+&varinteret;
if &varinteret=0 then nonevent=nonevent+1;
if incorrectevent=&effpasoutcome then do;problevel=0;output;end;
if last.problevel then output;
keep correctevent correctnonevent incorrectevent incorrectnonevent problevel;run;

/*we add a row*/
data sfdsf;put problevel correctevent correctnonevent incorrectevent  incorrectnonevent;run;
data plus;set sfdsf;
problevel=1;correctevent=0;correctnonevent=&effpasoutcome;incorrectevent=0;incorrectnon
event=&effoutcome;run;

data classif3;set classif2 plus;run;

/*we obtain sensibility and specificity*/
data discri;set classif3;retain sensitivity specificity x  y step 1 ;
sensitivity=correctevent/&effoutcome;specificity=correctnonevent/&effpasoutcome;
unmoinsspec=1-specificity;if problevel=0 then do;x=0;y=0;end;
if problevel=1 then do;x=1;y=1;end;step=1;run;
%mend;


/*Macro to draw the ROC curve*/
%macro roc;
title 'Plot of 2 ROC Curves on a Single Set of Axes';
title2 'Based on Data from a 1-5 Rating Scale';
filename gsf pipe 'lp -d printername';
goptions reset=all rotate=landscape
       gsfmode=replace gaccess=sasgastd gsfname=gsf cback=white;
```

```
*************************************************************************;
title1 height=2.0 font=swiss 'Figure 1';
title2 height=1.5 font=swiss 'ROC Curve' ;
symbol1 color=black font=swissx interpol=join line=1;
symbol2 color=black font=swissx interpol=join line=2;
axis1 label=(height=2.0 font=swiss 'False Positive Rate (1- Specificity)')
      value=(height=1.5 font=swiss ) order=(0.0 to 1.0 by 0.1) minor=none
      length=15 cm offset=(1,1)
      ;
axis2 label=(height=2.0 font=swiss 'Sensibility' rotate=90)
      value=(height=1.5 font=swiss) length=15 cm
      minor=none order=(0.0 to 1.0 by 0.1) offset=(0,0);

proc gplot data=discri;
plot sensitivity*(unmoinsspec)=1 x*y=1  /overlay haxis=axis1 vaxis=axis2;
footnote ' ' h=3;run;
%mend;


/*Macro to calculate the Area Under Curve*/
%macro auc;
proc sort data =discri;by descending problevel;run;

data try4;set discri;by step;
sensmoins1=lag(sensitivity);specmoins1=lag(unmoinsspec);
if first.step then do;sensmoins1=0;specmoins1=0;end;run;

/*calculation*/
data calculauctotale;set try4;by step;retain roc 0;
roc=roc+(0.5*((sensitivity+sensmoins1)*(unmoinsspec-specmoins1)));
if last.step then output;keep roc ;call symput('auc',roc);run;

%put "AREA UNDER CURVE WAS &auc";
%mend;


/*Macro to summarize results*/
%macro summary;
data kio;set tablesortiehosmer;if group=10 then call symput('chi2hl',somtot);run;
data kio;set tablesortiehosmer;if group=10 then output;keep somtot;run;

data fdsf;merge kio calculauctotale;run;
/*We adapt conclusions in function of results*/
data fsdf;
set fdsf;
calib=0;disc=0;
if 0<somtot<15.5 then do;calib=1;end;
if 0.80<=roc<=1 then do;disc=1;end;
call symput('calib2',calib);call symput('disc2',disc);
run;
```

```
%put %upcase("summary of results");
%put %upcase();
%put %upcase("The value of Khi2 Hosmer-Lemeshow was &chi2hl");
%put %upcase("Area under curve was &auc");
%put %upcase();

%if &calib2=1 %then %do;
	%put %upcase("The model was well calibrated");
%end;
%else %do;
	%put %upcase("The model was not well calibrated");
%end;

%if &disc2=1 %then %do;
	%put %upcase("The model correctly discriminated groups");
%end;
%else %do;
	%put %upcase("The model did not correctly discriminate groups");
%end;


%put %upcase();
%put %upcase(you can see graphics in the results section);
%mend;
```