La simulation d'un test adaptatif basé sur le modèle de Rasch
27 février 2002

Gilles Raîche
Université de Moncton
Faculté des sciences de l'éducation, Pavillon Jeanne-de-Valois
Moncton, NB EIA 3E9
Téléphone : (506) 858-4425
Télécopieur : (506) 858-4317
Courriel : raicheg@umoncton.ca

La simulation d'un test adaptatif basé sur le modèle de Rasch

Revue mesure et évaluation en éducation

Résumé

Les recherches sur le testing adaptatif reposent fréquemment sur des études de type Monte Carlo. Ces études permettent d'analyser les caractéristiques de la distribution d'échantillonnage des estimateurs en testing adaptatif et de provoquer des situations idéales, difficiles à réaliser dans la réalité de tous les jours. Il est alors possible de contrôler de façon plus efficace plusieurs variables. Intéressante, cette méthodologie est toutefois peu abordable puisque les programmes permettant la simulation de tests adaptatifs sont difficilement accessibles. Nous introduisons ici le code source commenté d'un tel programme en langage SAS.

Abstract

Monte Carlo methodologies are frequently applied to study the sampling distribution of the ability estimate in adaptive testing. With this approach we can learn about the sampling distribution of the ability estimate without constraints of real situations. Interesting, these Monte Carlo methodologies are not supported by the current availability of software programs, and when these programs are available their flexibility is limited. Here, a commented computer program coded in SAS language is proposed.

Introduction

À notre connaissance, aucun programme informatique n'est actuellement disponible pour permettre de simuler l'administration de tests adaptatifs. En fait, les chercheurs qui ont étudié les caractéristiques de tels tests produisent généralement eux-mêmes, et pour eux-mêmes, ces programmes : ceux-ci sont alors difficiles d'accès à la communauté scientifique. De plus, lorsque disponibles, ils sont peu flexibles et ne sont pas accompagnés du code source de façon à favoriser des adaptations.

Pour pallier cette situation et soutenir la recherche relative au testing adaptatif, nous proposons ici un tel programme accompagné de son code source en langage SAS. Il s'agit en fait d'un programme utilisé auparavant par Raîche (2000) pour étudier la distribution d'échantillonnage de l'estimateur du niveau d'habileté dans un test adaptatif en fonction des règles d'arrêt auquel a été apportées des améliorations. Ces améliorations donnent plus de flexibilité au programme originel et proposent de nouvelles méthodes d'estimation provisoire du niveau d'habileté. La modélisation de la réponse à l'item retenue est la modélisation logistique à un paramètre, soit le modèle de Rasch. Il serait facile de modifier le programme pour appliquer des modélisations à deux, trois ou quatre paramètres. La méthode d'estimation des estimateurs provisoire et final du niveau d'habileté retenue est la méthode de l'espérance a posteriori (EAP). Les nouvelles méthodes d'estimation provisoire proposées correspondent toutes à des modifications de la méthode de l'espérance a posteriori.

Une autre particularité de ce programme est qu'il permet de comparer les valeurs théoriques de l'erreur-type, de l'asymétrie et de la kurtose associées à chaque valeur obtenue de l'estimateur du

niveau d'habileté aux valeurs empiriques calculées à partir de chacune des distributions d'échantillonnage de l'estimateur du niveau d'habileté.

Initialisation des variables

À l'intérieur de la première section du programme, certaines variables doivent être spécifiquement initialisées dès le départ. Ainsi, le nombre maximal d'items administrés (item) et le nombre de sujets (sujet) doivent être déterminés à l'avance par l'utilisateur. Si le nombre maximal d'items administrés dépasse 60, il est recommandé d'augmenter le nombre de points de quadrature (qpoint) puisque les routines de calcul de l'asymétrie et de la kurtose nécessitent l'utilisation d'un plus grand nombre de points de quadrature que dans le cas des estimations usuelles en testing adaptatif. Raîche (2000, p. 136-147) fourni explications et indications quant au nombre de points de quadrature à utiliser.

Dans la plupart des cas, les autres valeurs n'ont pas à être modifiées. Par exemple, la constante de Haley (D) doit restée fixée à 1,702 pour assurer que les probabilités cumulatives obtenues par la fonction de probabilité logistique ne diffèrent pas plus de 0,01 avec celles obtenues par la loi normale. Il est toutefois possible de modifier l'intervalle d'intégration (thetamin, thetamax), l'estimateur a priori (baye) ou les valeurs du niveau d'habileté qui sont visées par la simulation (minhab, maxhab, byhab).

Cependant, une des originalités de ce programme est d'offrir l'opportunité d'étudier des procédures d'estimation proposées par Raîche et Blais (2002), elles-mêmes adaptatives : valeur adaptative de l'estimateur a priori (aap), correction adaptative du biais selon la formule proposée

par Bock et Mislevy en 1982 (acb) ou intervalle adaptatif d'intégration (aii). Il est aussi possible d'appliquer de façon conventionnelle la correction de Bock et Mislevy uniquement à la fin du test (fcb). Lorsque les variables globales aap, acb, aii ou fcb sont fixées à 0, le test adaptatif se déroule de façon usuelle, sans utiliser ces stratégies. Pour les appliquer, il faut initialiser à 1 chacune des stratégies qu'on désire étudier.

Fonctions

La macro %Prior permet de calculer la probabilité qu'un sujet affiche un niveau d'habileté spécifique lorsque la moyenne de la distribution a priori est fixée à la valeur de l'estimateur a priori du niveau d'habileté (baye) et que l'écart-type est égal à 1. La loi normale (baye, 1) est utilisée.

La macro %Logist sert à calculer la probabilité d'apparition d'une réponse à l'item (0 pour une mauvaise réponse, 1 pour une bonne réponse) selon la modélisation à un paramètre de la réponse à l'item. C'est la macro %Reponse qui génère de façon aléatoire une bonne réponse ou une mauvaise réponse à l'item en fonction de l'estimateur provisoire du niveau d'habileté (Harwell, Stone, Hsu et Kirisci, 1996, p. 116).

Pour déterminer le pas d'intégration à partir du nombre de points de quadrature qpoint et de l'intervalle d'intégration (thetamin - thetamax), la macro %Step est appelée par la macro %Mu2. C'est à partir de cette dernière que sont calculés le premier moment non centré, soit l'estimateur du niveau d'habileté, ainsi que les deuxième, troisième et quatrième moments centrés. Ces trois derniers moments non centrés correspondent respectivement à l'écart-type, à l'asymétrie et à la

kurtose de la distribution d'échantillonnage de l'estimateur du niveau d'habileté. Les équations permettant l'estimation de ces statistiques sont décrites par Raîche (2000), ainsi que par Blais et Raîche (2002 ; Raîche et Blais, 2002).

Enfin, les macros %Sitem et %Sseo permettent d'analyser les résultats de la simulation en fonction de la règle d'arrêt à étudier : selon le nombre d'items administrés avec %Sitem ou selon l'erreur-type de l'estimateur du niveau d'habileté avec %Sseo.

Résultats du programme

À titre d'exemple, nous présentons les résultats obtenus lors de la simulation de 500 administrations d'un test adaptatif qui ont été réalisées à un niveau d'habileté de -3 et à chacune des 7 valeurs de l'erreur-type retenue pour la règle d'arrêt (de 0,50 à 0,20 par sauts de 0,05) au tableau 2. Les résultats obtenus à chacune des valeurs du nombre d'items administrés (de 1 à 40 par sauts de 1), lorsque la règle d'arrêt selon le nombre d'items est appliquée, sont présentés au tableau 3. Auparavant, au tableau 1, nous décrivons les variables produites en sortie par le programme.

Tableau 1. Variables produites en sortie par le programme SIMTAO 1.1

RSEO	Erreur-type retenue pour la règle d'arrêt (lorsque %RSEO est utilisé)
ITEM	Nombre d'items administrés retenu pour la règle d'arrêt (lorsque %RITEM est utilisé)
MTEO	Moyenne des valeurs obtenues de l'estimateur du niveau d'habileté pour %SUJET sujets
SE	Erreur-type empirique calculée sur les valeurs obtenues de l'estimateurs du niveau d'habileté
MSEO	Moyenne des valeurs théoriques obtenues de l'erreur-type de l'estimateur du niveau d'habileté
AS	Asymétrie empirique calculée sur les valeurs obtenues de l'estimateurs du niveau d'habileté
MASO	Moyenne des valeurs théoriques de l'asymétrie de l'estimateur du niveau d'habileté
KU	Kurtose empirique calculée sur les valeurs obtenues de l'estimateurs du niveau d'habileté
MKUO	Moyenne des valeurs théoriques de la kurtose de l'estimateur du niveau d'habileté

Tableau 2. Simulation d'un test adaptatif en fonction de la règle d'arrêt selon l'erreur-type

---------- HABILETE=-3 ----------

OBS	RSEO	SE	MSEO	MTEO	AS	MASO	KU	MKUO
1	20	0.19574	0.19843	-2.86735	0.11396	-0.04698	0.25201	0.049934
2	25	0.25252	0.24701	-2.80753	0.27885	-0.07072	0.17099	0.066383
3	30	0.31028	0.29521	-2.73056	0.30850	-0.09111	-0.04540	0.067628
4	35	0.36666	0.34223	-2.63158	0.40457	-0.10454	0.17772	0.058355
5	40	0.41216	0.38983	-2.50978	0.31117	-0.11697	-0.04393	0.048892
6	45	0.41746	0.43603	-2.36559	0.59112	-0.14301	0.62322	0.059110
7	50	0.47513	0.48159	-2.18776	0.22684	-0.15117	-0.83537	0.047787

Tableau 3. Simulation d'un test adaptatif en fonction de la règle d'arrêt selon le nombre d'items administrés

---------- HABILETE=-3 ----------

OBS	ITEM	MTEO	SE	MSEO	AS	MASO	KU	MKUO
1	1	-0.55234	0.12336	0.82605	8.99056	-0.10337	79.1467	0.12845
2	2	-0.96589	0.14864	0.72751	6.75061	-0.18018	44.6226	0.16464
3	3	-1.29344	0.18587	0.66402	4.65296	-0.21728	21.1136	0.16994
4	4	-1.55309	0.23565	0.61625	2.89965	-0.22737	7.6644	0.15817
5	5	-1.77271	0.27796	0.57780	2.02508	-0.22464	3.1483	0.13351
6	6	-1.94580	0.31936	0.54350	1.36880	-0.21067	0.8764	0.10341
7	7	-2.09104	0.34974	0.51299	0.93989	-0.19322	-0.0559	0.07467
8	8	-2.21253	0.37128	0.48554	0.71883	-0.17540	-0.0066	0.05362
9	9	-2.31790	0.37892	0.46109	0.52809	-0.16050	-0.0362	0.04202
10	10	-2.39150	0.37568	0.43810	0.37986	-0.14888	-0.0532	0.04436
11	11	-2.44573	0.37099	0.41748	0.37734	-0.14064	0.0139	0.04928
12	12	-2.51386	0.36117	0.40056	0.31944	-0.13420	0.1032	0.04622
13	13	-2.55526	0.35980	0.38407	0.27246	-0.12506	0.0225	0.04579
14	14	-2.59567	0.35359	0.36956	0.32091	-0.12045	-0.0922	0.04448
15	15	-2.63012	0.34918	0.35602	0.31528	-0.11480	-0.0375	0.04594
16	16	-2.65362	0.33411	0.34373	0.34075	-0.11288	0.0050	0.05310
17	17	-2.67329	0.33009	0.33202	0.27598	-0.10729	-0.1235	0.05489
18	18	-2.68974	0.32909	0.32113	0.24310	-0.10172	-0.1183	0.05702
19	19	-2.70290	0.31428	0.31132	0.31374	-0.10050	0.0152	0.06323
20	20	-2.72380	0.30870	0.30270	0.33443	-0.09818	0.0436	0.06402

SIMTAO 1.1 : code source (SAS 6.0)

```
/*************************************************************************/;
/*************************************************************************/;
/**                                                                     **/;
/**                     Simulation d'un test adaptatif                  **/;
/**                                                                     **/;
/**                             SIMTAO 1.1                              **/;
/**                           Gilles Raîche                             **/;
/**                                                                     **/;
/**                      Université de Moncton                          **/;
/**                    Faculté des sciences de l'éducation              **/;
/**  Département d'enseignement au secondaire et des ressources humaines **/;
/**                                                                     **/;
/**                            Février 2002                             **/;
/**                                                                     **/;
/*************************************************************************/;
/*************************************************************************/;

 Options replace nodate pageno=1 pagesize=256 nosource nonotes
         nomtrace nomlogic nomprint;

/* Initialisation des constantes ****************************************/;
 %Let nitem=10   ;  /* Nombre d'items                                    */;
 %Let qpoint=40  ;  /* Nombre de points de quadrature                    */;
 %Let D=1.702    ;  /* Constante de Haley                                */;
 %Let thetamin=-4;  /* Limite inférieure d'intégration dans EAP          */;
 %Let thetamax= 4;  /* Limite supérieure d'intégration dans EAP          */;
 %Let bayes=0    ;  /* Niveau d'habileté a priori                        */;
 %Let sujet=100  ;  /* Nombre de valeurs différentes du niveau d'habileté  */;
 %Let minhab= -3 ;  /* Niveau d'habileté minimal                         */;
 %Let minhab=  3 ;  /* Niveau d'habileté maximal                         */;
 %Let byhab=   1 ;  /* Sauts entre le niveau d'habileté minimal et maximal */;
 %Let aap=0      ;  /* Bascule pour estimateur a priori adaptatif        */;
 %Let acb=0      ;  /* Bascule pour correction pour le biais adaptative  */;
 %Let fcb=0      ;  /* Bascule pour correction pour le biais finale      */;
 %Let ain=0      ;  /* Bascule pour intervalle d'intégration adaptatif   */;
/* *********************************************************************/;

 TITLE1 "SIMULATION D'UN TEST ADAPTATIF (n=&SUJET)";
 TITLE2 "(ESTIMATION ADAPTATIVE : AAP=&AAP, AIN=&AIN, ACB=&ACB, FCB=&FCB)";

 %Macro Prior (Theta,mu);
 %* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
 %* Fonction de probabilité a priori N(mu,1) telle qu'utilisée par Bock  *;
 %* et Mislevy (1982)                                                   *;
 %* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
 %* Variables locales  : theta, mu                                      *;
 %* Variables globales : step                                           *;
 %* Appel de fonction  : aucun                                          *;
 %* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
  Probnorm(&Theta-&mu + step/2) - Probnorm(&Theta-&mu - step/2);;
 %Mend;

 %Macro logist (Theta,b,D,r);
 %* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
 %* Fonction logistique à un paramètre                                  *;
 %* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
 %* Variables locales  : theta, b, r                                    *;
 %* Variables globales : D                                              *;
 %* Appel de fonction  : aucun                                          *;
```

```
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
 ((((1+exp(-&D*(&Theta-&b)))**-1))**&r) *
 (1-(((1+exp(-&D*(&Theta-&b)))**-1)))**(1-&r);
%Mend;

%Macro Reponse (i,habilete);
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
%* Génération d'une réponse à un item en fonction du niveau d'habileté     *;
%* préalablement et du niveau de difficulté du prochain                    *;
%* item selon la méthode utilisée par Harwell, Stone, Hsu et Kirisci (1996)*;
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
%* Variables locales  : i, habilete                                        *;
%* Variables globales : D, b                                               *;
%* Appel de fonction  : aucun                                              *;
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
 plog= exp(&D*(&habilete-b{&i})) / (1+exp(&D*(&habilete-b{&i})));
 puni=ranuni(1);
 r{&i}=plog > puni;
%Mend;

%Macro step;
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
%* Calcul de la largeur de classe en fonction du nombre de points de       *;
%* quadrature ainsi que de la surface de la somme de la fonction a priori  *;
%* sur l'intervalle Thetamin … Thetamax                                    *;
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
%* Variables locales  : surface, j                                         *;
%* Variables globales : step, thetamax, thetamin, qpoint                   *;
%* Appel de fonction  : prior                                              *;
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
 surface=0;
 Do j=1 to &nitem;
    %* Largeur de classe en fonction du nombre de points de quadrature .....;
    step=(&thetamax-&thetamin)/(&qpoint-1);
    end;
 %* Calcul de la surface sous la fonction a priori .......................;
 Do theta=&thetamin to &thetamax by step;
    surface=surface + %Prior(theta,0);
    end;
 %* Put "surface = " surface; *;
%Mend;

%Macro MU2(habilete);
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
%* Estimation des moments par la méthode EAP                               *;
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
%* Variables locales  : b1,j,num,p,lnp,i,logist,prior                      *;
%* Variables globales : nitem,habilete,thetamin,thetamax,step,theta,b,D,r  *;
%*                      bayes,te,denom                                     *;
%* Appel de fonction  : reponse,logist,prior                               *;
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
 %* Paramètre de difficulté du premier item ..............................;
 b1=0;
 %* Initialisation à 0 des numérateurs et dénominateurs ...................;
 %* ainsi que de l'estimateur a priori ....................................;
 Do j=1 to &nitem;
    num(j)=0;denom(j)=0;bayes(j)=&bayes;
    num2_(j)=0;num3_(j)=0;num4_(j)=0;
    end;
 %* Estimation des j niveaux d`habileté ...................................;
 Do j=1 to &nitem;                                        %* Début niveau 1 .;
```

```
    %* Paramètres adaptatifs pour l'intégration numérique ..................;
    ain=0;If &ain=1 then do; ain=b(j); end;
    %Reponse(j,&habilete);
    %* Calcul de l'espérance a posteriori du niveau d'habileté .............;
    Do theta=(&thetamin+ain) to (&thetamax+ain) by step; %* Début niveau 2 .;
       p=1;lnp=log(p);
       Do i=1 to j;                                      %* Début niveau 3 .;
          logist=%logist(theta,b(i),&D,r(i));
          lnp=lnp+log(logist);
          end;                                           %* Fin niveau 3 ...;
       if j>1 then do; prior=%prior(theta,bayes(j-1));end;
       if j=1 then do; prior=%prior(theta,&bayes);end;
       num(j)=num(j)+     (theta)*exp(lnp+log(prior));
       denom(j)=denom(j)+  exp(lnp+log(prior));
       end;                                              %* Fin niveau 2 ...;
    %* jième estimation du niveau d`habileté ...............................;
    te(j)=num(j)/denom(j);


    Do theta=(&thetamin+ain) to (&thetamax+ain) by step; %* Début niveau 2 .;
       p=1;lnp=log(p);
       Do i=1 to j;                                      %* Début niveau 3 .;
          logist=%logist(theta,b(i),&D,r(i));
          lnp=lnp+log(logist);
          end;                                           %* Fin niveau 3 ...;
       if j>1 then do; prior=%prior(theta,bayes(j-1));end;
       if j=1 then do; prior=%prior(theta,&bayes);end;
       num2_(j)=num2_(j)+((theta-te(j))**2)  *exp(lnp+log(prior));
       num3_(j)=num3_(j)+((theta-te(j))**3)  *exp(lnp+log(prior));
       num4_(j)=num4_(j)+((theta-te(j))**4)  *exp(lnp+log(prior));
       end;                                              %* Fin niveau 2 ...;
    %* jième estimation de l'erreur type du niveau d'habileté ..............;
    se(j)=(num2_(j)/denom(j))**.5;
    %* jième estimation de l'asymétrie ......................................;
    as(j)=(num3_(j)/denom(j))/se(j)**3;
    %* jième estimation de la kurtose .......................................;
    ku(j)=((num4_(j)/denom(j))/se(j)**4)-3;
    denom(j)=denom(j)/surface;
    %* Correction adaptative de l'estimateur a posteriori ..................;
    if &acb=1 then do; te(j)=te(j)/(1-(se(j)*se(j))); end;
    %* Estimation adaptative de l'estimateur a priori ......................;
    if &aap=1 then do; bayes(j) = te(j);end;
    if j<&nitem then do;
       b(j+1) =te(j);
       end;

    end;                                                 %* Fin niveau 1 ...;
%Mend;




%MACRO SITEM;
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
%* Calcul des valeurs théoriques et empiriques des estimateurs en fonction *;
%* de différentes valeurs de la règle d'arrêt selon le nombre d'items      *;
%* administrés                                                             *;
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
%* Variables locales  : aucune                                             *;
%* Variables globales : sujet, nitem                                       *;
%* Appel de fonction  : aucune                                             *;
```

```
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
Proc means data=thesd_&nitem noprint;
     class habilete item;
     var seo aso kuo;
     output out=res1 mean(seo)=mseo mean(aso)=maso mean(kuo)=mkuo;
     run;
Proc sort;by habilete item;run;

Proc means data=thesd_&nitem noprint;
     class habilete item;
     var teo;
     output out=res2 mean=mteo std=se skewness=as kurtosis=ku;
     run;
Proc sort;by habilete item;run;

Data results;
     merge res1 res2;
     by habilete item;
     if (habilete NE "" and item NE "");
     run;
Proc sort;by habilete;run;

Proc print;
     by habilete;
     var item mteo se mseo as maso ku mkuo;
     run;
%MEND;


%Macro SSEO;
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
%* Calcul des valeurs théoriques et empiriques des estimateurs en fonction *;
%* de différentes valeurs de la règle d'arrêt selon l'erreur-type          *;
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
%* Variables locales  : h, i, j                                             *;
%* Variables globales : sujet, nitem, minhab, maxhab, byhab                 *;
%* Appel de fonction  : aucune                                              *;
%* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *;
 %DO H=50 %TO 20 %BY -5;
   %DO I=&minhab %TO &maxhab %BY &byhab;
     %DO J=1 %TO &SUJET;
       Data seo40; set thesd_&nitem ;
            If (habilete=&I and sujet=&J and seo<=(&H/100)) then do;
               rseo=&H;
               output seo40;
               stop;
               end;
               Run;
       Proc datasets nolist; append base=sorti data=seo40;run;
     %END;
   %END;
 %END;

 Proc means data=sorti noprint;
      class habilete rseo;
      var  seo aso kuo;
      output out=res1 mean(seo)=mseo mean(aso)=maso mean(kuo)=mkuo;
      run;
 Proc sort;by habilete rseo;run;

 Proc means data=sorti noprint;
```

```
      class habilete rseo;
      var teo;
      output out=res2 mean=mteo std=se skewness=as kurtosis=ku;
      run;
 Proc sort;by habilete rseo;run;

 Data results;
      merge res1 res2;
      by habilete rseo;
      if (habilete NE "" and rseo NE "");
      run;
 Proc sort;by habilete;run;

 Proc print;
      by habilete;
      var rseo se mseo mteo as maso ku mkuo;
      run;
 Proc datasets;delete sorti;run;
%MEND;


Data work.thesd_&nitem;
     array te[&nitem];     /* Estimation de l'habileté ...................*/;
     array r[&nitem];      /* Vecteur de réponses ........................*/;
     array b[&nitem];      /* Paramètres de difficulté ...................*/;
     array num[&nitem];    /* Numérateur dans EAP (MU) ...................*/;
     array num2_[&nitem];  /* Numérateur dans EAP (SE) ...................*/;
     array num3_[&nitem];  /* Numérateur dans EAP (MC3) ..................*/;
     array num4_[&nitem];  /* Numérateur dans EAP (MC4) ..................*/;
     array denom[&nitem];  /* Dénominateur dans EAP ......................*/;
     array se[&nitem];     /* Erreur type ................................*/;
     array as[&nitem];     /* Asymétrie ..................................*/;
     array ku[&nitem];     /* Kurtose ....................................*/;
     array prc[&nitem];    /* Proportion de bonnes réponses ..............*/;
     array bayes[&nitem];  /* Estimateur a priori ........................*/;
     length default=3      /* Format de storage des variables ............*/;
     b1=0;                 /* Initialisation du niveau de difficulté de b1*/;
     %STEP;
     Do sujet = 1 to &sujet;                            %* Début niveau 1 .;
        Do habilete = &minhab to &maxhab by &byhab;     %* Début niveau 2 .;
        %MU2(habilete);
        /* Valeurs des différentes statistiques selon le nombre ..........*/;
        /* d'items administrés ............................................*/;
        Do item=1 to &nitem;                            %* Début niveau 3 .;

          ro=r{item};bo=b{item};teo=te{item};prco=prc{item};baye=bayes{item};
          seo=se{item};aso=as{item};kuo=ku{item};
          if &fcb=1 then do;teo=teo/(1-seo*seo);end;
          biais = teo-habilete;
          Output;
          end;                                          %* Fin niveau 3 ...;
        End;                                            %* Fin niveau 2 ...;
        End;                                            %* Fin niveau 1 ...;
     Keep sujet item ro bo habilete teo prco seo aso kuo biais baye;
     run;


%SSEO;
%SITEM;
```

Références

Blais, J.G. et Raîche, G. (2002). *Some features of the estimated sampling distribution of the ability estimate in computerized adaptive testing according to two stopping rules*. Communication au 11e Biennal intenational objective measurement workshop. New Orleans : IOMW.

Bock, D.R. et Mislevy, R.J. (1982). Adaptive EAP estimation of ability in a micro computer environment. *Applied psychological measurement*, *6*(4), 431-444.

Harwell, M.R., Stone, C.A., Hsu, T.C. et Kirisci, L. (1996). Monte Carlo studies in item response theory. *Applied psychological measurement*, *20*(2), 101-125.

Raîche, G. (2000). *La distribution d'échantillonnage de l'estimateur du niveau d'habileté en testing adaptatif en fonction de deux règles d'arrêt : selon l'erreur-type et selon le nombre d'items administrés*. Thèse de doctorat inédite. Montréal : Université de Montréal.

Raîche, G. et Blais, J.G. (2002). Étude de la distribution d'échantillonnage de l'estimateur du niveau d'habileté en testing adaptatif en fonction de deux règles d'arrêt dans le contexte de l'application du modèle de Rasch. *Mesure et évaluation en éducation*.

Raîche, G. et Blais, J.G. (2002*). Practical considerations about expected a posteriori estimation in adaptive testing : adaptive a priori, adaptive correction for bias, and adaptive integration interval*. Communication au 11e Biennal intenational objective measurement workshop. New Orleans : IOMW.