

SAS**Travaux Dirigés****IV. Le macro-langage****Références Bibliographiques**

- *SAS, Maîtriser SAS Base et SAS Macro*, H. Kontchou-Kouomegni & O. Decourt, Chapitre 7, éditions Dunod.
- *SAS : Introduction au décisionnel*, S. Ringuedé, Chapitre 9, éditions Pearson.

Pour toute question, commentaire ou suggestion, contacter : salim.lardjane@univ-ubs.fr

- 1) Commençons par simuler, à l'aide du programme qui suit, la consommation de 10 marchandises par les différents membres de sept familles pendant 11 ans (données annuelles).

```
data tab;
  input famille $ @@;
  membre = round(ranuni(54)*5+1);
  do nummbr = 1 to 7;
    if ranuni(5652) > 0.5 then sexe = 1;
    else sexe = 0;
    do annee = 1997 to 2007;
      if nummbr <= membre then output;
    end;
  end;
cards;
Dupuis Fournier Durand Dufour Martin Dupont Moreau
;
run;

data tab;
  set tab;
  array conso{10} conso1-conso10;
  do i = 1 to 10;
    conso(i) = round(abs(rannor(52))*ranuni(545)*100,1);
  end;
run;

proc print data = tab;
run;
```

Expliquer le fonctionnement du programme précédent.

- 2) Ecrire un programme permettant de représenter graphiquement (par un diagramme en barres) la différence entre la consommation moyenne du bien 1 par les différentes familles et la consommation moyenne globale du bien 1.

Commencer par compléter la table par une colonne dont toutes les lignes contiennent la consommation moyenne globale du bien 1.

Définir ensuite une variable contenant la différence entre la consommation de bien 1 par chaque individu et la consommation moyenne globale de bien 1.

Enfin, utiliser la **proc sgplot** pour obtenir le diagramme en barres souhaité.

- 3) Le code utilisé est le suivant.

```
data tab;
    set tab;
    one = 1;
run;

proc sort data = tab;
    by one;
run;

proc means data = tab mean;
    var consol;
    output out = mout mean = mconsol;
    by one;
run;

data tab;
    merge tab mout;
    by one;
    diff_consol = consol-mconsol;
run;

proc sgplot data = tab;
    title "Différence de consommation moyenne du bien 1 par famille";
    yaxis label = "Consommation 1";
    vbar famille / response = diff_consol stat = mean;
run;
```

Modifier le programme précédent de façon à pouvoir obtenir le graphique précédent pour le bien 2. A cet effet, on introduira une macro-variable **c** contenant le numéro du bien (ici, 2).

- 4) Le code utilisé est le suivant.

```
%let c = 2;
proc sort data = tab;
    by one;
run;

proc means data = tab mean;
    var conso&c;
    output out = mout mean = mconso&c;
    by one;
run;

data tab;
    merge tab mout;
```

```

    by one;
    diff_conso&c = conso&c-mconso&c;
run;

proc sgplot data = tab;
    title "Différence de consommation moyenne du bien &c par famille";
    yaxis label = "Consommation &c";
    vbar famille / response = diff_conso&c stat = mean;
run;

```

La procédure consistant à créer une table de résultats (**mout**) pour ensuite la fusionner avec la table de départ n'est en fait pas efficace. En effet, elle conduit à alourdir la table de départ en y ajoutant des variables qui ne servent qu'à des calculs intermédiaires ; de plus, la procédure de fusion peut être très lente lorsque les tables sont de grandes tailles.

Simplifier le programme précédent en sauvegardant la consommation globale moyenne du bien dans une macro-variable et en l'utilisant pour calculer la différence entre consommation du bien et consommation globale moyenne du bien pour chaque individu.

5) Le code utilisé est le suivant.

```

%let c = 2;

proc sort data = tab;
by one;
run;

proc means noprint data = tab mean;
    var conso&c;
    output out = mout mean = mconso&c;
    by one;
run;

data _null_;
    set mout;
    call symput("moy",mconso&c);
    call symput("moytr",round(mconso&c,0.1));
run;

data tab;
    set tab;
    diff_conso&c = conso&c-&moy;
    %put La moyenne pour la variable conso&c est &moy
run;

proc sgplot data = tab;
    title "Différence de consommation moyenne du bien &c par famille";
    title2 "La consommation moyenne des individus en bien &c est de
        %left(&moytr)";
    yaxis label = "Consommation &c";
    vbar famille / response = diff_conso&c stat = mean;
run;

```

On cherche à présent à représenter graphiquement (par des diagrammes en barres), pour chaque année, *l'écart entre la consommation moyenne des différentes familles en un bien donné et la consommation moyenne constatée sur la même année pour le même bien pour toutes les familles de l'échantillon.*

Proposer un programme permettant d'obtenir le résultat voulu. Celui-ci devra être conçu de façon à ce que la variable-clef **année** puisse être remplacée facilement par la variable **sexe**, par exemple.

6) Le code utilisé est le suivant.

```
%let c = 2;
%let clef = annee;

proc sort data = tab;
by &clef;
run;

proc means noprint data = tab mean;
var conso&c;
output out = mout mean = m&clef.conso&c;
by &clef;
run;

data _null_;
set mout;
clee = 'a' || left(&clef);
call symput(clee, m&clef.conso&c);
run;

data tab;
set tab;
clee = 'a' || left(&clef);
varr = symget(clee);
diff_conso&clef&c = conso&c - varr;
run;

proc sgplot data = tab;
title "Différence de consommation moyenne du bien &c par famille";
title2 "pour chaque &clef";
yaxis label = "Consommation &c";
vbar famille / response = diff_conso&clef&c stat = mean;
by &clef;
run;
```

On souhaite à présent rédiger un macro-programme permettant de construire automatiquement une table des cours journaliers de neuf actions et du CAC 40 depuis le 1^{er} janvier 2010 à partir de données récupérées sur <http://fr.finance.yahoo.com> et sauvegardées dans le répertoire .../yahoo. Ce site offre en effet la possibilité de télécharger des historiques des cours journaliers de différents titres, le début de la période de l'historique dépendant du titre. Par exemple, le lien <http://ichart.yahoo.com/table.csv?s=ALU.PA> permet de télécharger l'intégralité des données journalières disponible pour l'action Alcatel Lucent (ALU).

Rédiger un programme SAS permettant de récupérer les données concernant le cours d'ouverture de l'action Alcatel Lucent (fichier .../yahoo/ALU.PA) dans une table SAS appelée alcatel.

7) Le code utilisé est le suivant.

```
ods graphics on;
options helpbrowser = sas /* Win 64 */;
```

```
filename alcatelb "d:/data/sasdata/yahoo/alu.pa";
data alcatel;
  infile alcatelb;
  input date ddmmyy10. cours;
  format date ddmmyy10.;
run;
proc print data = alcatel;
run;
```

Pour construire la table complète, il faut récupérer les cours d'ouverture des neuf actions à suivre. Commencer par définir, à l'aide d'une **étape data**, des macro-variables act1, ..., act9 contenant les codes des actions, des macro-variables lactAC, ..., lactCA contenant les noms exacts des sociétés correspondantes et une macro-variable num contenant le nombre d'actions suivies.

Les codes des actions et les noms des sociétés correspondantes sont donnés ci-dessous.

AC	Accor
ACA	Crédit Agricole
AF	Air France - KLM
AI	Air Liquide
ALO	Alstom
ALU	Alcatel Lucent
BN	Danone
BNP	BNP - Paribas
CA	Carrefour

8) Le code utilisé est le suivant.

```
data act;
  input code $ libel $20.;
  call symput (compress('act' || _n_), code);
  call symput (compress('lact' || code), libel);
  call symput ('num', _n_);
cards;
AC      Accor
ACA     Crédit Agricole
AF      Air France - KLM
AI      Air Liquide
ALO     Alstom
ALU     Alcatel Lucent
BN      Danone
BNP     BNP - Paribas
CA      Carrefour
;
```

Modifier le programme de lecture de la table en remplaçant les éléments variables d'une table à l'autre par des macro-variables et en faisant intervenir le numéro de l'action dans le tableau précédent (6 pour Alcatel Lucent). On appellera c_act6 la table obtenue et on pourra utiliser la fonction SAS **strip** pour récupérer le code de l'action.

9) On obtient par exemple le résultat suivant.

```
filename actt6 "d:/data/sasdata/yahoo/%sysfunc(strip(&act6)).pa";
data c_act6;
    infile actt6;
    input date ddmmyy10. &act6;
    format date ddmmyy10.;
run;
proc print data = c_act6;
run;
```

Modifier et compléter ce programme à l'aide d'un **%do** itératif de façon à obtenir un macro-programme permettant de récupérer l'information concernant les 9 actions suivies. Appeler le macro-programme obtenu.

10) Le code utilisé est le suivant.

```
%macro lecture;
    %do i = 1 %to &num;
        filename actt&i "d:/data/sasdata/yahoo/%sysfunc(strip(&&act&i)).pa";
        data c_act&i;
            infile actt&i;
            input date ddmmyy10. &&act&i;
            format date ddmmyy10.;
        run;
    %end;
%mend lecture;

%lecture;
```

Nous disposons à présent de 9 tables qu'il faut réunir au sein d'une même table SAS nommée base. On se limitera à la période 1^{er} janvier 2004 – date du jour courant. Rédiger un macro-programme permettant d'obtenir la table base. On définira une variable date qu'on utilisera comme clef de fusion.

11) Le code utilisé est le suivant.

```
%macro fusion;
    data base;
        do date = "01jan2004"d to "&sysdate9"d;
            output;
        end;
    run;

    %do i = 1 %to &num;
        proc sort data = c_act&i;
            by date;
        run;
        data base;
            merge base c_act&i;
            by date;
        run;
    %end;
%mend fusion;

%fusion;

proc print data = base;
run;
```

On souhaite à présent compléter la table base avec l'évolution du CAC 40. Lire les données .../yahoo/CAC.PA dans une table nommée cac et la fusionner avec la table base avec la date comme variable-clef.

12) Le code utilisé est le suivant.

```
filename cac "d:/data/sasdata/yahoo/CAC.pa";
data cac;
  infile cac;
  input date ddmmyy10. cac;
  format date ddmmyy10.;
run;
proc sort data = cac;
  by date;
run;
data base;
  merge base cac;
  by date;
run;
proc print data = base;
run;
```

Supprimer de la table base les observations antérieures au 1^{er} janvier 2004 et celles correspondant à des samedi ou des dimanche (pas de quotation). On pourra utiliser la **fonction put** dans la définition d'une variable indiquant le jour de la semaine (instruction **put(date,weekdatx9.)**).

13) Le code utilisé est le suivant.

```
data base;
  set base;
  if date < '01jan2004'd then delete;
  ctrl = strip(put(date,weekdatx9.));
  if ctrl = 'Saturday' or ctrl = 'Sunday' then delete;
run;
proc print data = base;
run;
```

La **proc expand** permet d'interpoler les valeurs manquantes de séries chronologiques. L'utiliser pour interpoler les valeurs manquantes de la table base.

14) Le code utilisé est le suivant.

```
proc expand data = base out = base;
  id date;
run;
proc print data = base;
run;
```

On souhaite à présent calculer les rendements des différentes actions et du CAC 40. Si on note X_t la valeur de la série à la date t , le rendement à la date t est défini par

$$r_t = \frac{X_t - X_{t-1}}{X_{t-1}}$$

Créer une variable `r_cac` contenant le rendement du CAC 40 pour les différentes date de la période considérée.

15) Le code utilisé est le suivant.

```
data base;
  set base;
  r_cac = dif(cac)/lag(cac);
run;
proc print data = base;
run;
```

En modifiant le programme précédent, écrire un macro-programme permettant de créer 9 variables contenant les rendements des actions suivies pour les différentes date de la période considérée.

16) Le code utilisé est le suivant.

```
%macro rendements;
  %do i = 1 %to &num;
    data base;
      set base;
      r_&&act&i = dif(&&act&i)/lag(&&act&i);
    run;
  %end;
%mend rendements;

%rendements;

proc print data = base;
run;
```

On souhaite utiliser les données obtenues pour construire un graphique représentant l'évolution du rendement de deux actions au choix parmi les neuf précédentes ainsi, éventuellement, que celui du CAC 40, à partir d'une date au libre choix de l'utilisateur.

Rédiger un macro-programme permettant d'obtenir ce résultat. On fera bien attention à ramener les rendements en base 100 au début de la période à considérer. On pourra utiliser la macro-fonction `%scan` et passer les noms des actions à représenter en paramètres mots-clefs du macro-programme.

17) Le code utilisé est le suivant.

```
%macro graphique (action = bnp aca ,cac = non, datedebut = 01jan2007);
  %let date = %sysfunc(inputn(&datedebut,date9.));
  %let dateb = %sysfunc(putn(&date,fradfwdx.));
  %let datefoot = %sysfunc(today(),fradfwkx.);

  %let act1 = %scan(&action,1);
  %let act2 = %scan(&action,2);

  /* calcul des cours des actions et du cac en base 100 à la date de début */

  data base;
    set base;
    drop b100;;
  run;

  data base;
    set base;
```



```

        if date <= "&datedebut"d then b100cac = 100;
        if date > "&datedebut"d then b100cac+b100cac*r_cac;
run;

%do i = 1 %to 2;
    data base;
        set base;
        retain b100&act&i;
        if date <= "&datedebut"d then b100&act&i = 100;
        if date > "&datedebut"d then b100&act&i =
            b100&act&i+b100&act&i*r_&act&i;
    run;
%end;

/* affichage des graphiques */

%if &cac = oui %then %do;
    proc sgplot data = base;
        yaxis display = (nolabel);
        series y = b100&act1 x = date;
        series y = b100&act2 x = date;
        series y = b100cac x = date;
        label b100&act1 = "&lact&act1";
        label b100&act2 = "&lact&act2";
        label b100cac = "cac 40";
        where date >= "&datedebut"d;
        title;
        title2 h = 12pt "Evolution comparée des cours du cac et des
            actions";
        title3 h = 12pt "%sysfunc(strip(&lact&act1)) et
            %sysfunc(strip(&lact&act2))";
        title4 h = 12pt "Base 100 au &dateb";
        footnote      j = left "Graphique généré le &datefoot"
            j = right "Source des données : Yahoo Finance";
    run;
%end;

%if &cac = non %then %do;
    proc sgplot data = base;
        yaxis display = (nolabel);
        series y = b100&act1 x = date;
        series y = b100&act2 x = date;
        label b100&act1 = "&lact&act1";
        label b100&act2 = "&lact&act2";
        where date >= "&datedebut"d;
        title;
        title2 h = 12pt "Evolution comparée des cours des actions";
        title3 h = 12pt "%sysfunc(strip(&lact&act1)) et
            %sysfunc(strip(&lact&act2))";
        title4 h = 12pt "Base 100 au &dateb";
        footnote      j = left "Graphique généré le &datefoot"
            j = right "Source des données : Yahoo Finance";
    run;
%end;
%mend graphique;

%graphique(action = alo bn ,cac = oui, datedebut = 01sep2010);

```