

Dans ce T.P., plusieurs éléments nouveaux de SAS vont être abordés : tout d'abord, la procédure `princomp`, permettant de réaliser une Analyse en Composantes Principales (A.C.P.); ensuite, la notion de table d'annotations (*annotate data set*), permettant d'améliorer certains graphiques, indispensable en A.C.P.; enfin, la manipulation d'une table SAS obtenue comme sortie d'une procédure (en l'occurrence `princomp`).

Exercice 06.1

SAS dispose de la procédure `princomp` pour réaliser l'A.C.P. usuelle (voir pages 67–68). Mettre cette procédure en œuvre, sur les données de températures, à l'aide des commandes suivantes :

```
proc princomp data=sasuser.temp;
run;
```

Qu'obtient-on comme résultats? Commenter.

Faire ensuite :

```
proc princomp data=sasuser.temp
              vardef=n cov
              out=acp1 outstat=acp2;
```

```
run;
```

Puis :

```
proc print data=acp1; run;
```

Faire enfin :

```
proc print data=acp2; run;
```

Comparer les sorties standard de la procédure `princomp` avec le contenu des tables `acp1` et `acp2`.

Exercice 06.2

Copier les fichiers suivants :

```
~baccini/tpsas/exolim/data/class.don
```

```
~baccini/tpsas/exolim/data/class.txt
```

Construire une table SAS contenant les données de `class.don` (prendre les noms de variables indiqués dans `class.txt`) et contrôler.

Comparer ensuite les résultats des deux séries d'instructions ci-dessous (la deuxième fait appel à une table d'annotations; voir pages 44).

1.

```
proc gplot data=sasuser.class;
  plot weight*height;
  run;
  quit;
```
2.

```
data exo2;
  set sasuser.class;
  x=height; /* x est le paramètre désignant l'axe horizontal */
  y=weight; /* y est celui désignant l'axe vertical */
  xsys='2'; /* la valeur 2 permet d'avoir les étiquettes à l'emplacement des points */
  ysys='2';
  text=name; /* name est la variable alphanumérique dont les valeurs s'afficheront */
  size=1; /* la valeur fixe la taille des caractères */
  keep x y text xsys ysys size; /* pour alléger la table créée */
  run;
  proc print data=exo2; run;
```

```

proc gplot data=exo2;
plot y*x / annotate=exo2;
* ce n'est que ci-dessus que exo2 prend le statut de table d'annotations;
run;
quit;

```

Pour comprendre le rôle des paramètres, tester ensuite successivement les valeurs suivantes :
`xsys='3' ysys='3' size=3 size=0.5.`

Exercice 06.3

Sous UNIX, dans votre répertoire `sasuser`, copier la table SAS suivante :
`~baccini/sasuser/arch/crime.sas7bdat`

(on crée ainsi la table `sasuser.crime` correspondant à un exemple de la bibliothèque SAS).

En vous inspirant des 2 exercices précédents, réaliser l'A.C.P. centrée et réduite de cette table SAS et représenter les individus libellés dans le premier plan principal.

On notera que les différentes options de la procédure `gplot` déjà rencontrées (`symbol`, `axis...`) peuvent être utilisées pareillement avec une table d'annotations.

Remarques :

- De façon à ne pas utiliser la variable quantitative `STATE`, sans intérêt, rajouter, dans la procédure `princomp`, la commande :
`var MURDER RAPE ROBBERY ASSAULT BURGLARY LARCENY AUTO;`
- On notera que `MURDER` représente les meurtres, `RAPE` les viols, `ROBBERY` les vols avec violence, `ASSAULT` les agressions, `BURGLARY` les cambriolages, `LARCENY` les vols sans violence (larcins) et `AUTO` les vols d'automobiles.

Exercice 06.4

L'objet de cet exercice est de récupérer les valeurs propres de l'A.C.P. en colonnes dans une table SAS. Pour cela, copier le fichier

`~baccini/tpsas/exolim/exo06_4.sas`

puis l'exécuter séquence par séquence, pour bien comprendre le fonctionnement du programme. Commenter (on pourra se reporter aux exemples 2.3 du cours photocopié, pages 18–19).

Exercice 06.5

De façon similaire, créer une table SAS, que vous appellerez `tvectp`, contenant en lignes les vecteurs propres; transposez ensuite cette table pour obtenir une nouvelle table SAS, appelée `vectp`, dont les colonnes correspondent aux vecteurs propres (la séquence `do; _name_='lambda'; end;` est ici inutile).

Exercice 06.6

On souhaite maintenant réaliser l'éboulis des valeurs propres de l'A.C.P. On rappelle que l'éboulis des valeurs propres est un graphique comportant en abscisses les dimensions de l'A.C.P. (1, 2, 3...) et en ordonnées les valeurs propres correspondantes.

Pour obtenir ce graphique, la table `lambda` doit avoir été créée (exercice 06.4). Suivre ensuite les instructions suivantes :

```

data eboul;
set lambda;
input k @@;      /* 11 */
cards;          /* 12 */
1 2 3 4 5 6 7
;
run;

```

```
proc print;
run;
proc gplot data=eboul;
plot valpr*k;
run;
quit;
```

/* 11 */ Création de la variable **k** au sein de la table **eboul** ;
le double symbole “@” (à *commercial*, ou *arrobas*, se dit “at”) permet de lire les données en ligne.
/* 12 */ Saisie des données “à la main”.

Exercice 06.7

Reprendre l’exercice précédent pour représenter l’éboulis des valeurs propres de telle façon que les points soient joints par un trait continu et représentés de façon plus nette ; enlever les subdivisions inutiles de l’axe des abscisses ; faites apparaître “dimensions” sur l’axe des abscisses et “valeurs propres” sur celui des ordonnées.

Exercice 06.8

L’objet de cet exercice est d’étudier diverses possibilités de la procédure **print** de SAS, en particulier en réutilisant des commandes telles que **select** ou **when ... output**.

Copier le fichier :

~baccini/tpsas/exolim/exo06_8.sas.

Le faire exécuter et commenter.