

L'objet de ce T.P. est de se familiariser avec les procédures de base de SAS relatives aux variables quantitatives et de réaliser les premiers graphiques. Le support est constitué des chapitres 2 et 3 du cours polycopié.

Dans chacun des 3 exercices considérés, on réutilisera le fichier de données `notes.don`.

Exercice 02.1

1. Entrer dans SAS, faire exécuter le programme suivant

```
proc print data=notes; run;
```

 et commenter.
2. Construire maintenant la table SAS, appelée `sasuser.notes`, contenant les données du fichier `notes.don`. La faire afficher dans la fenêtre *output* et contrôler (on pourra réutiliser le programme `exo01_2.sas`).
3. Sur la table SAS ainsi obtenue, utiliser les procédures `means` (page 34), `univariate` (page 33), `corr` (page 34–35) et `plot` (page 36).
4. Récupérer les résultats sous forme de fichier ASCII et les faire imprimer (essayer de contrôler les options d'impression).

Exercice 02.2

On s'intéresse ici à la procédure `rank` et à son fonctionnement. Exécuter successivement les séquences suivantes et commenter au fur et à mesure. Essayer de bien comprendre la gestion des tables SAS par la procédure `rank`.

```
proc rank data=sasuser.notes;
run;
/* ----- */
proc print;
run;
/* ----- */
proc print data=sasuser.notes;
run;
/* ----- */
proc rank data=sasuser.notes out=rangs;
run;
proc print data=rangs;
run;
/* ----- */
proc rank data=sasuser.notes out=rangs;
var MECA ALIN;
proc print data=rangs;
run;
/* ----- */
proc rank data=sasuser.notes out=rangs;
var MECA ALIN;
ranks RMECA RALIN;
proc print data=rangs;
run;
```

Exercice 02.3

1. Refaire la procédure `corr` avec l'option `cov` ; commenter les résultats obtenus.
2. Utiliser la procédure `standard` (pages 20–21) pour réduire les 5 variables considérées.
Indications : au niveau de la gestion des tables SAS, la procédure `standard` est analogue à la procédure `rank` ; les options `mean=0` et `std=1` sont obligatoires ; rajouter également l'option `vardef = n`.
3. Faire maintenant la procédure `corr` sur les variables réduites avec les options `cov` et `vardef = n`. Commenter.
4. Faire la procédure `sort` (page 23) avec la commande `by` (commande obligatoire) sur l'une des variables initiales. Faire ensuite afficher le résultat dans la fenêtre `output`, sachant que la procédure `sort` est encore analogue, en ce qui concerne la gestion des tables, aux procédures `rank` et `standard`.
5. Recommencer en rajoutant l'option `descending` entre la commande `by` et le nom de la variable. Commenter.

Remarques complémentaires

- Conventions de langage (conformes au cours polycopié). Nous appellerons `option` toute directive se plaçant entre le nom d'une procédure SAS et le point-virgule marquant la fin de la première ligne de cette procédure. Nous appellerons `commande` toute directive se plaçant au-delà de ce point-virgule et avant la ligne `run` marquant la fin de la procédure. On notera aussi que certaines commandes (par exemple `by`) comportent elles-même des options (par exemple `descending`).
- Pour une synthèse sur la notion d'option, on se reportera aux remarques complémentaires de la feuille de T.P. 03.
- L'option `descending`, utilisée avec la procédure `sort`, peut également être utilisée, dans les mêmes conditions, avec la procédure `rank`.
- On notera que certaines procédures élémentaires de SAS affichent directement les résultats dans la fenêtre `output` (`means`, `univariate`, `corr`...). D'autres, au contraire, ne produisent aucun résultat si elles ne sont pas suivies de la procédure `print` (`rank`, `standard`, `sort`...).
- Relancer un programme SAS quelconque en le faisant commencer par la ligne suivante :
`dm 'clear log'; dm 'clear output';`
(`dm` est mis pour *display manager*).
Que se passe-t-il ?
- Consulter le cours polycopié (page 101) pour voir l'utilité des commandes `ps` et `kill` d'UNIX. Ces commandes peuvent s'avérer nécessaires si l'on se retrouve bloqué dans SAS.