IMIS : Master 1
Université Paris Est – Marne la Vallée

TP 1. Introduction au logiciel SAS
Analyse Statistique Univariée

1. Premier contact avec SAS

1. Lancez le logiciel sas. Vous voyez apparaître les fenètres Editeur (Editor), Journal (Log), Sortie (Output), Résultats (Results) et Explorateur (Explore).
2. Dans la fenètre "SAS : Program Editor" entrez le programme suivant :

```
DATA TP1; /* creation d'une table provisoire */
INPUT Taille Poids Sexe $;
CARDS;
174 65 M
169 56 F
166 48 F
181 80 M
168 53 F
176 76 M
190 77 M
159 70 F
162 60 F
164 51 F
160 73 F
RUN;
PROC PRINT;
RUN;
```

3. Sauvegardez le fichier
   `Fichier > Enregistrer sous... > progtp1.sas`
4. Exècutez le programme
   `Exécuter > Soumettre` ou la touche F3
   Remarque : On peut également sélection une partie du programme de la fenètre "Editeur", et l'exécuter en choisissant l'option "soumettre la sélection" après un clique droite sur le texte sélectionné.
   Pour sauvegarder aprés avoir modifié :

`Fichier > Enregistrer`

5. Pour vérifier le contenu de la table créée :
   `Outils > Editeur de tables SAS | Fichier > Ouvrir | Work > TP1 > Ouvrir`
   On remarquera que ce menu peut servir non seulement à la visualisation des tables existantes, mais aussi à la création de nouvelles tables.
   Remarque : On peut également visualiser la table créée en accédant, dans la fenètre "Explorateur", à la "bibliothèque" puis à "Work" et enfin en ouvrant le fichier "TP1"
6. Aller dans les onglets Outils/option/preferences et cocher "html" dans le menu de sortie des résultats pour un affichage plus joli des résultats.
7. Ajouter l'option `NOOBS` à la procédure `PRINT`. Que fait elle ?

Remarques importantes

1) SAS ne différencie pas les majuscules et les minuscules. Par exemple, on peut très bien écrire `PRINT DATA=tp1` pour visualiser les données contenues dans la table `TP1`.
2) Ne pas oublier les `;` à la fin de chaque instruction.
3) Si après avoir exécuté le programme vous obtenez un résultat bizarre, vérifiez la fenêtre SAS : Log.
4) Le signe "dollar" dans la déclaration des variables indique que la variable précédant `$` est qualitative.
5) Dans un programme SAS, tout ce qu'on écrira entre `/*` et `*/` ne sera pas pris en compte pendant l'exécution du programme. Ceci sert à commenter différentes parties du programme.
6) Sauvegardez toujours votre programme avant de le soumettre.

2. Personnalisation et raccourcis

1. **Création d'une librairie de travail.** Par défaut ; les données entrées dans SAS sont enregistrées dans la librairie "WORK" qui est effacée à chaque fois que l'on quitte le logiciel. Pour conserver le données, il est recommandé de créer une librairie (un répertoire) qui sera conservée entre différentes sessions SAS. Ainsi, pour créer la librairie TPSAS, il suffit de rajouter une option globale au programme en première ligne :
   `LIBNAME TPSAS '?/TPSAS' ; (? designe le chemin qui mene au repertoire)`
   Cette étape n'est réalisée qu'une fois au début de la session. Pour indiquer à SAS que l'on souhaite enregistrer les données dans cette librairie, il faut précéder le nom des données par "TPSAS." (à chaque création de données). Ainsi, la ligne DATA ... du programme *progtp1.sas* deviendra
   `DATA TPSAS.TP1 ;`
   Si, au cours d'une session, le préfixe "TPSAS." est omis, les données seront enegistrées dans la librairie par défaut WORK.

2. Il est souvent difficile de lire le rapport d'erreur qui apparait dans la fenêtre LOG lorsque les rapports s'accumulent. Pour eviter ce problème on peut simplement faire
`Editeur > Effacer tout`
dans la fenêtre LOG.
3. Enfin, vous aurez sans doute remarqué le nombre grandissant de fenêtres. Pour retrouver vos fenêtres JOURNAL, EDITEUR et SORTIE, les touches F. peuvent vous aider :
F5= EDITEUR, F6=JOURNAL et F7=SORTIE.
Sinon, on peut toujours choisir dans le menu
`Affichage > {Editeur, Journal, Sortie}`

3. ANALYSE STATISTIQUE UNIVARIÉE

0. Pour calculer les caractéristiques statistiques les plus élémentaires (moyenne, écart type, variance, min, max, ...) d'une variable, on peut utiliser la procédure `MEANS`. On ne la testera pas car la procédure `UNIVARIATE` traité ci-après est plus générale.

1.a. La procédure `SORT` permet de trier les données ; elle range par défaut les données quantitatives en ordre croissant et les données qualitatives en ordre alphabétique. Afin d'obtenir l'ordre inverse, il faut intercaler l'option `DESCENDING` apres `BY` .

Pour tester cette procédure, on ajoute dans notre programme les lignes suivantes :

```
PROC SORT DATA=TP1;
BY Sexe;
RUN;
PROC PRINT;
RUN;
```

1.b. Pour ne pas écraser la table "TP1", on peut créer une nouvelle table qui contiendra les données triées. Pour cela, il faut utiliser l'option `OUT=ma_lib.TP1_triee` dans la procédure `SORT`.

2. La procédure `RANK` calcule les rangs de variables quantitatives. Sa syntaxe est
PROC RANK <options> ;
BY <descending> variable ; /* si on veut trier selon la variable */
RANKS liste de nouvelles variables ; /* contiendra les rangs */
VAR liste de variables ; /* les variables dont on calcule le rang */
Les options les plus importantes de cette procédure sont
- *data=table sas* indique le nom de la table, par défaut la dernière créée,
- *out=table sas* spécifie le nom de la table créée qui contiendra les variables initiales et les rangs,
- *descending* rangs par valeurs décroissantes.

Les instructions les plus importantes de cette procédure sont

BY suivi du nom d'une variable qualitative indique que les statistiques sont calculées par groupe d'observations, cette instruction ne peut être appliquée qu'aux données triées (cf. la procédure SORT).

RANKS doit être spécifiée si l'on veut que les variables initiales soient recopiées en sortie,

VAR les rangs des variables de cette liste sont calculés ; par défaut toutes les variables quant. sont traitées.

On appliquera cette procédure à la variable taille en ordre décroissant et groupé par sexe.

```
PROC RANK DATA=TP1 OUT=RANGS;
VAR Taille Poids;
RANKS VAR1 VAR2;
BY Sexe;
RUN;
```

Pour vérifier le résultat :

```
PROC PRINT DATA=RANGS;
RUN;
```

3. Lecture des fichiers extérieurs : Ouvrez un éditeur de text quelconque ('Bloc-Notes' marche bien) et entrez les données taille-poids-sexe. Enregistrez le fichier dans un répertoire x sous le nom TP1.dat. Ajouter à la fin une colonne contenant les données :

   ```
   20  25  24  26  25  27  33  24  26  23  31.
   ```

   Enregistrez.

   Afin de lire ces données dans un programme SAS, on utilise la commande `INFILE` de la procédure DATA :

   ```
   DATA TP1;
   INFILE 'x/TP1.dat';
   INPUT Taille Poids Sexe $ Age;
   RUN;
   ```

4. Pour illustrer la procédure `UNIVARIATE`, saisissez et exécutez le programme suivant :

   ```
   DATA TP1;
   INFILE 'x/TP1.dat';
   INPUT Taille Poids Sexe $ Age;
   RUN;
   OPTIONS LINESIZE=132 PAGESIZE=66 NODATE;
   FOOTNOTE 'TP1 : Procedure UNIVARIATE';
   PROC UNIVARIATE NORMAL PLOT;
   VAR TAILLE;
   BY SEXE;
   RUN;
   ```

Afin d'enregistrer certaines des statistiques calculées dans une table extérieure, on peut utiliser l'instruction (en l'insérant par exemple entre BY SEX et RUN),

```
OUTPUT out=univar N=nbObs MEAN=moyenne USS=CarresObs KURTOSIS=CoeffApllat;
```

Faites la même chose sans spécifier la commande `BY`.

On remarque que les mots NORMAL et PLOT qui suivent la procédure UNIVARIATE sont des options. La première permet d'obtenir des tests de normalité, alors que la seconde dessine des graphiques.

On peut également spécifier une variable qui contient les pondérations des observations. Pour cela, il faut rajouter l'instruction `WEIGHT variable`.

5. La procédure `PLOT` permet de dessiner des graphiques en basse résolution de nuages de points en deux dimensions.

```
PROC PLOT DATA=TP1;
BY SEXE;
PLOT TAILLE*POIDS='*';
RUN;
```

Dans le cas où on a plus de deux variables quantitatives, par exemple `Taille Poids` et `Age`, on peut demander dans une seule commande les graphiques des nuages de points `Taille*Poids` et `Poids*Age`. Cela se fait comme suit :

```
PLOT TAILLE*POIDS='*' POIDS*AGE='+';
```

et si l'on veut les superposer

```
PLOT TAILLE*POIDS='*' POIDS*AGE='+' / OVERLAY;
```

6. Pour obtenir des graphiques plus jolis, on utilise les graphiques haute résolution. Les procédures les plus souvent utilisées sont `GPLOT` et `GCHART`.

Après avoir appelé une procédure de graphique haute résolution, il faut absolument la quitter en utilisant la commande `QUIT;`.

```
PROC GPLOT DATA=TP1;
    SYMBOL1 v=square interpol=r c=black;
    SYMBOL2 v=plus interpol=rcclm c=black;
PLOT TAILLE*POIDS=1;
RUN;
QUIT;
```

Faites la même chose en remplaçant `PLOT TAILLE*POIDS=1;` par `PLOT TAILLE*POIDS=2;`

ANNEXE : STATISTIQUES CALCULÉES PAR **UNIVARIATE**

N – le nombre d'observations,
MEAN – la moyenne empirique,
SUM OBSERVATIONS – la somme des observations,
STD DEVIATION – standard deviation (écart type) mais divisé par $n-1$,
VARIANCE – la variance,
SKEWNESS – le coefficient d'asymétrie,
KURTOSIS – le coefficient d'aplatissement,
UNCORRECTED SS –la somme des carrés des observations,
CORRECTED SS –la somme des observations centrées par la moyenne empirique,
COEFF VARIATION – $(s/\bar{X}) \cdot 100\%$,
STD ERROR MEAN – $s/\sqrt{n}$,
RANGE – l'étendue de l'échantillon ($\max - \min$),
INTERQUARTILE RANGE – l'écart interquartile,
STUDENT'S T – la statistique $t = \bar{X}\sqrt{n}/s$ (pour tester $\mu = 0$),
SIGN M – $(N^+ - N^-)/2$, où N^+ est le nombre d'observation > 0, N^- est le nombre d'observation < 0. (pour tester $Med = 0$),
SHAPIRO-WILK – $(\sum a_i X_{(i)})^2/ns^2$,
KOLMOGOROV-SMIRNOV – $\max |i/n - F_i|$, où $F_i = \Phi((X_{(i)} - \bar{X})/s)$,
CRAMER-VON MISES – Kolmogorov mais somme au lieu de max, mieux si il y a des observations aberrantes.
ANDERSON-DARLING – $A^2 = -N - \sum(2i-1)/N[\ln(\Phi(X_{(i)}) + \ln(1 - \Phi(X_{(N+1-i)})]$,

Pour tous les tests : si la p-value est petite (disons < 0.05), on rejette l'hypothèse nulle et on accepte l'alternative. Dans le cas contraire (p-value ≥ 0.05), on accepte l'hypothèse nulle H_0.