Manipulation des Fichiers de Données sous SPPS

1. Création d'un Fichier de Travail.

La plupart du temps, les fichiers de données des grosses enquêtes comportent plusieurs centaines de variables. Or, dans le cadre d'un travail de recherche on peut être amené à ne travailler que sur certaines variables.

Par exemple, vous faites une étude sur la santé. Vos données sont situées dans le fichier *sante.sav* qui contient les variables suivantes :

Nom	**Libellé**
id	Identifiant
carnet	Possède un carnet de santé
consult	A consulté un médecin au cours des 6 derniers mois
datecons	Date de la dernière consultation
vacchb	Vacciné contre hépatite B
vaccha	Vacciné contre hépatite A
vaccroug	Vacciné contre la rougeole
vaccgrip	Vacciné contre la grippe
sida1	A déjà entendu parler du sida
sida2	A déjà utilisé un préservatif
sida3	A déjà fait un test de dépistage

Votre recherche porte sur la vaccination. Vous n'avez donc pas besoin des variables *sida1*, *sida2* et *sida3*. Il y a alors tout intérêt à créer un sous fichier de données que nous appellerons *travail.sav* qui ne contiendra que les variables qui nous seront utiles. Lorsque l'on travaille sur de très grosses bases de données, cela permet à l'ordinateur de fonctionner beaucoup plus rapidement.

Pour cela, il faut ouvrir le fichier *sante.sav* dans l'éditeur de données, puis ouvrir l'éditeur de syntaxe.

Nous allons utilisé la commande *SAVE OUTFILE* dont la syntaxe est détaillée dans l'Encadré 1 :

Dans notre cas, nous souhaitons ne pas garder les variables *sida1*, *sida2* et *sida3*. La syntaxe correspondante sera donc :

```
SAVE OUTFILE='travail.sav'
/DROP  sida1
       sida2
       sida3.
EXECUTE.
```

Encadré 1 SAVE OUTFILE

```
SAVE OUTFILE='nom_du_fichier.sav'

        /DROP   var1
                var2
                var3

        /KEEP   var11
                var12
                var13

        /RENAME
                anc_var21=nouv_var21
                anc_var22=nouv_var22
.
```

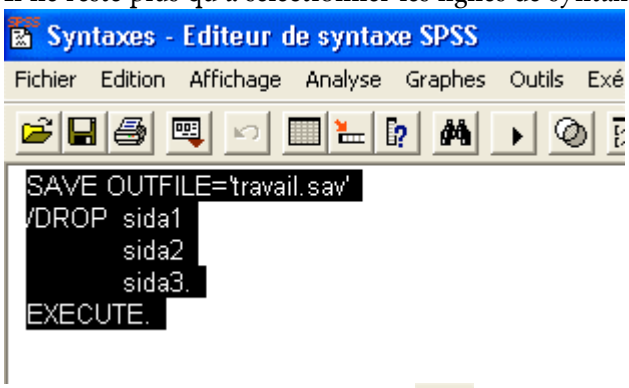
nom_du_fichier.sav est le nom du fichier qui sera créé. À défaut de précision, ce fichier sera placé dans le même répertoire que le fichier source.

Les options /DROP /KEEP et /RENAME sont optionnelles.

/DROP exclut les variables *var1*, *var2* et *var3* et garde toutes les autres variables.
/KEEP garde les variables *var11*, *var12* et *var13* et exclut toutes les autres variables.
/RENAME permet de renommer certaines variables au passage, *anc_21* est ainsi renommée *nouv_var21*.

ATTENTION : Ne pas oublier les points en fin de commande.

Il ne reste plus qu'à sélectionner les lignes de syntaxe :



et à appuyer sur le bouton flèche  (en ayant bien vérifier auparavant que le fichier chargé dans l'éditeur de données est *sante.sav*).

Or, la plupart du temps lorsque l'on travaille sur de grosses bases de données, le nombre de variable que l'on garde est inférieur à celui que l'on exclut. On utilisera alors préférentiellement l'option */KEEP*. Cela donnera la syntaxe suivante qui aboutit au même résultat.

```
SAVE OUTFILE='travail.sav'
/KEEP  id
       carnet
       consult
       datecons
       vacchb
       vaccha
       vaccroug
       vaccgrip.
EXECUTE.
```

2. Ajout de Variables.

Supposons les données ont été scindées en plusieurs fichier de données. Ainsi les caractéristiques individuelles sont situées dans un fichier *indiv.sav* alors que les variables sur la santé sont dans le fichier *travail.sav* que nous venons de créer.

Le fichier *indiv.sav* contient les variables suivantes :

Nom	**Libellé**
id	Identifiant
sexe	Sexe
age	Age
instruct	Niveau d'instruction
milieu	Milieu de résidence
matri	État matrimonial

Pour notre analyse, nous avons besoin des variables *sexe*, *age* et *milieu*. Nous avons donc besoin de les incorporer à notre base de données *travail.sav*. Pour cela, il est impératif que les deux fichiers de données aient le même identifiant, ce qui est la cas ici avec la variable *id*.

2.1 Trier les Observations.

Avant de commencer, il faut trier les observations des deux fichiers selon l'identifiant :

- Ouvrir le fichier *indiv.sav*.
- Cliquer dans le menu sur *Données* > *Trier les observations*..
- Sélectionner la variable *id* par ordre croissant.

- Cliquer sur *OK*.
- Enregistrer le fichier de données.
- Recommencer avec le fichier *travail.sav*.

Il est également possible d'avoir recours à la syntaxe suivante :

```
SORT CASES BY id (A).
EXECUTE.
```

Encadré 2 SORT CASES

SORT CASES BY *var1* (A *ou* D) *var2* (A *ou* D).

Cette commande trie les observations du fichier courant tout d'abord selon *var1* puis selon *var2* si deux observations ont la même valeur pour *var1*.

Entre parenthèses est indiqué par un A si le tri est ascendant (croissant) et par un D si le tri est descendant (décroissant).

Exemple :
SORT CASES BY nom (A) age (D). trie les observations selon le nom de manière croissante. Si plusieurs observations ont le même nom, elles sont triées selon l'âge de manière décroissante.

2.2 Ajouter les Variables.

Nous allons avoir recours à la commande *Données* > *Fusionner des fichiers* > *Ajouter des variables*... .

Encadré 3 Ajout de variables

Ajouter des variables permet de fusionner le fichier de données actif avec un fichier de données SPSS externe qui contient les mêmes observations mais pas les mêmes variables. Par exemple, vous souhaitez fusionner un fichier de données contenant les résultats d'un test préalable avec un autre fichier contenant les résultats d'un test final.

Les observations doivent être présentées dans le même ordre dans les deux fichiers.

Si une ou plusieurs clés d'appariement sont utilisées pour apparier les observations, les deux fichiers de données doivent présenter ces clés d'appariement par ordre croissant.

Les noms de variable du second fichier, redondants avec les noms du fichier actif, sont exclus par défaut. En effet, SPSS part du principe que ces variables contiennent des informations redondantes.

Variables exclues : Variables à exclure du nouveau fichier fusionné. Par défaut, cette liste contient tous les noms de variable du fichier externe qui sont redondants avec ceux du fichier actif. Les variables du fichier actif sont identifiées par un astérisque (*). Les variables du fichier externe sont identifiées par un signe (+). Si vous souhaitez inclure une variable exclue avec un nom redondant dans le fichier fusionné, vous pouvez le renommer puis l'ajouter à la liste des variables à inclure.

Variables du nouveau fichier actif : Variables à inclure dans le nouveau fichier actif. Par défaut, tous les noms de variables uniques dans les deux fichiers sont inclus dans la liste.
Clés d'appariement : Si certaines observations dans un fichier n'ont pas de concordance dans l'autre fichier (c'est-à-dire que certaines observations manquent dans un fichier), utilisez les clés d'appariement pour identifier et correctement apparier les observations des deux fichiers. Vous pouvez également utiliser ces clés avec des fichiers de consultation de table.

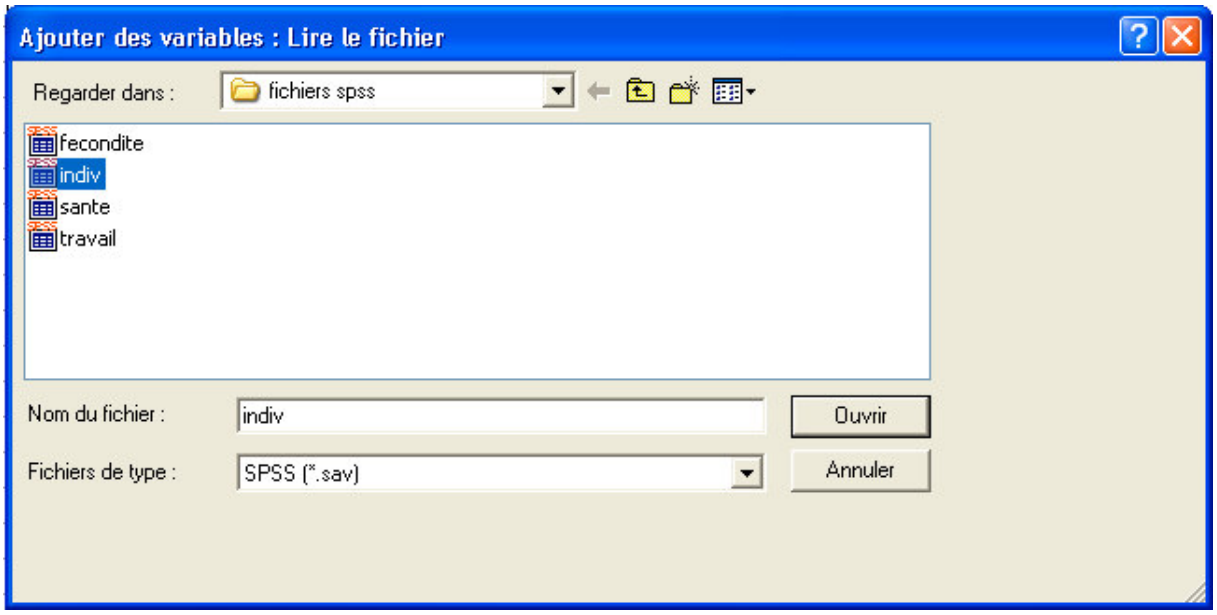
Les clés d'appariement doivent avoir le même nom dans les deux fichiers de données.

Les deux fichiers de données doivent présenter les variables par ordre croissant ou décroissant et l'ordre des variables de la liste clés d'appariement doit être le même que l'ordre de tri.

Les observations qui n'ont pas de correspondance dans les clés d'appariement sont inclus dans le fichier fusionné mais elles sont fusionnées avec les observations de l'autre fichier. Les observations sans correspondance contiennent des valeurs uniquement pour les variables du fichier duquel elles sont issues. Les variables de l'autre fichier contiennent la valeur manquante au système.

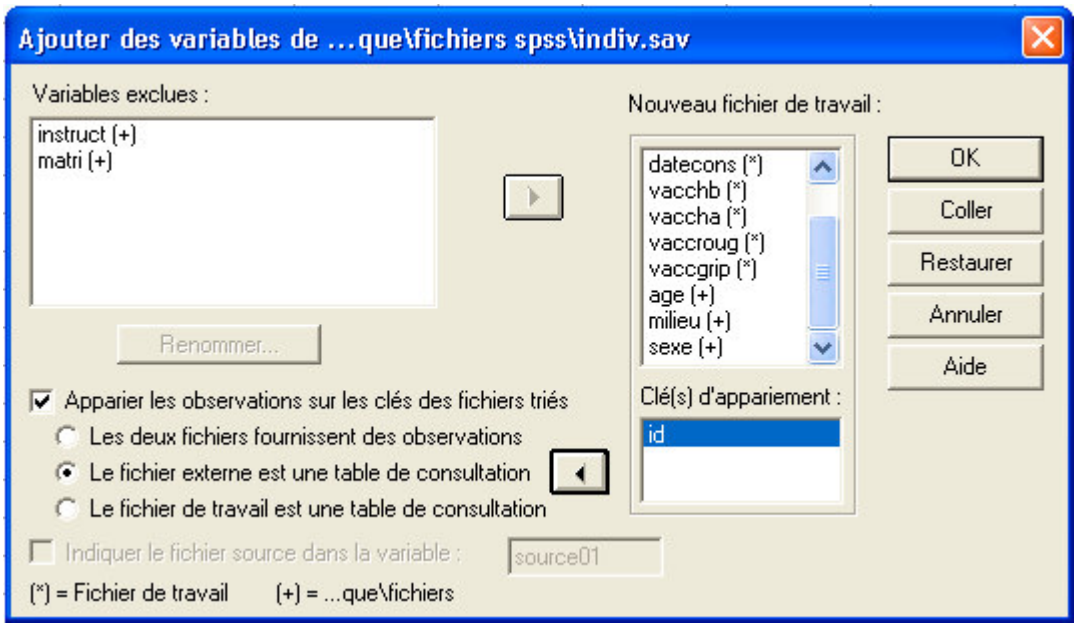
Le fichier externe ou fichier actif est une table codée : Une table codée ou un fichier de consultation de table, est un fichier dans lequel les données de " chaque observation " peuvent s'appliquer à plusieurs observations de l'autre fichier. Par exemple, si un des fichiers contient des informations sur les différents membres d'une famille (sexe, âge, niveau scolaire) et l'autre des informations générales sur la famille (revenu global, taille, habitat), vous pouvez utiliser le fichier sur la famille comme fichier de consultation et appliquer les informations générales à chaque membre de la famille dans le fichier fusionné.

Tout d'abord nous allons indiquer à SPSS qu'il doit aller chercher les variables supplémentaires dans le fichier *indiv.sav*.



Dans notre cas, nous souhaitons ajouter les variables *age*, *milieu* et *sexe*. Les autres variables du fichier *indiv.sav*, à savoir *instruct* et *matri* sont donc à exclure.

D'autre part, nous allons apparier les observations selon la variable *id* et considérer que le fichier externe est une table de consultation. Nous obtenons donc la boite de dialogue suivante :



Après avoir cliquer sur *OK*, SPSS nous rappelle que les observations des deux fichiers doivent être triées selon la clé d'appariement c'est-à-dire selon *id*. Si cela a été fait, cliquer sur *OK*, sinon sur *Annuler*, trier les deux fichiers et recommencer l'opération.

Remarque :

Si les questions sur la santé n'ont été posées qu'à certaines personnes de l'enquête et que le fichier *sante.sav* ne porte que sur ces personnes, alors il y aura plus d'observations dans *indiv.sav* que dans *sante.sav*. Si pour l'analyse, vous avez besoin de tenir compte des personnes à qui les questions sur la santé n'ont pas été posées, alors il vaut mieux cocher l'option *Les deux fichiers fournissent des observations*.

3. Création d'un Fichier Enfants.

Nous disposons également d'un fichier *fecondite.sav* comportant des données sur la fécondité des personnes enquêtées. Ce fichier comporte les variables suivantes :

Nom	**Libellé**
id	Identifiant
nbenf	Nombre d'enfants
sexe$1	Sexe enfant 1
annee$1	Année de naissance de l'enfant 1
dc$1	L'enfant 1 est-il décédé ?
sexe$2	Sexe enfant 2
annee$2	Année de naissance de l'enfant 2
dc$2	L'enfant 2 est-il décédé ?
sexe$3	Sexe enfant 3
annee$3	Année de naissance de l'enfant 3
dc$3	L'enfant 3 est-il décédé ?

Pour les besoins de l'analyse, on peut avoir besoin d'un fichier *enfants.sav* où chaque observation porte sur un enfant. (Le même problème se pose lorsque l'on veut créer un fichier de trajets à partir d'un fichier de parcours migratoires, ou un fichier de maladies à partir d'un fichier recensant les différentes maladies qu'a eu un individu…)

3.1 Création des Fichiers *enfant1.sav*, *enfant2.sav* et *enfant3.sav*.

Dans un premier temps, il nous faut créer des sous fichiers pour chaque rang de naissance, avant de fusionner ces différents sous fichiers en un seul.

Nous allons donc avoir recours à la commande *SAVE OUTFILE*. Nous allons garder la variable *id* qui va nous être indispensable par la suite, ainsi que la variable *nbenf*.
Dans le même temps, nous allons renommer *sexe$1* en *sexe*, *année$1* en *annee*, *dc$1* en *dc*, *sexe$2* en *sexe*, etc. afin de pouvoir procéder par la suite à la fusion des fichiers.

Nous allons donc écrire la syntaxe suivante :

```
SAVE OUTFILE='enfant1.sav'
        /KEEP     id
                  nbenf
                  sexe$1
                  annee$1
                  dc$1
        /RENAME
                  sexe$1=sexe
                  annee$1=annee
                  dc$1=dc
.
EXECUTE.

SAVE OUTFILE='enfant2.sav'
        /KEEP     id
                  nbenf
                  sexe$2
                  annee$2
                  dc$2
        /RENAME
                  sexe$2=sexe
                  annee$2=annee
                  dc$2=dc
.
EXECUTE.

SAVE OUTFILE='enfant3.sav'
        /KEEP     id
                  nbenf
                  sexe$3
                  annee$3
                  dc$3
        /RENAME
                  sexe$3=sexe
                  annee$3=annee
                  dc$3=dc
.
EXECUTE.
```

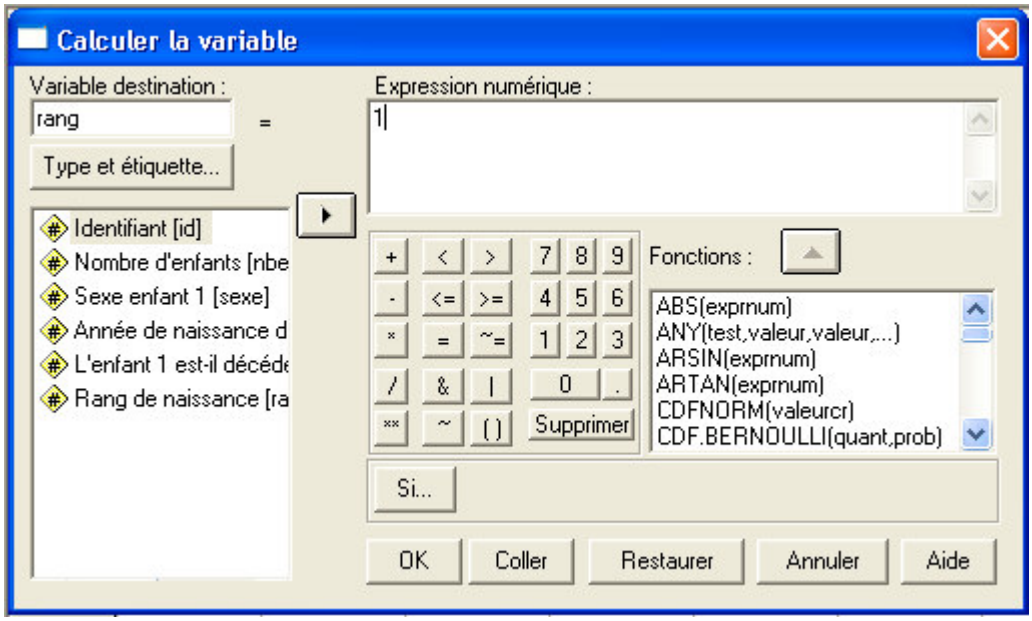
ATTENTION : avant d'exécuter la commande, il faut que le fichier actif soit bien *fecondite.sav*.
D'autre part, une erreur courante consiste à oublier les points en fin de commande.

Astuce :
Pour gagner en temps, on peut écrire la syntaxe pour le premier fichier, puis faire un copier/coller et enfin avoir recours à la commande *Édition* > *Remplacer* afin de remplacer le 1 en 2, etc. Cependant, il convient d'être vigilant et de bien relire l'ensemble de la syntaxe avant de l'exécuter.

3.2 Création de la Variable Rang de Naissance.

Une information importante est le rang de naissance qui était fourni par l'extension $. Il nous faut donc fournir cette information que nous n'avons plus.
Nous allons donc créer une variable *rang* dans chacun des sous fichier, puisque nous savons que chaque sous fichier contient des enfants de même rang.

- Ouvrir le fichier *enfant1.sav*.
- Cliquer sur *Transformer* > *calculer...* .
- Mettre rang dans *Variable destination*.
- Mettre 1 dans *Expression numérique*.
- Cliquer sur *OK*.
- Sauvegarder le fichier.
- Recommencer l'opération avec *enfant2.sav* en remplaçant 1 par 2 dans *Expression numérique*, etc.



Toute cette opération peut-être réalisée en une fois avec la syntaxe suivante :

```
GET FILE='enfant1.sav'.
COMPUTE rang = 1 .
EXECUTE .
SAVE OUTFILE='enfant1.sav'.

GET FILE='enfant2.sav'.
COMPUTE rang = 2 .
EXECUTE .
SAVE OUTFILE='enfant2.sav'.

GET FILE='enfant3.sav'.
COMPUTE rang = 3 .
EXECUTE .
SAVE OUTFILE='enfant3.sav'.
```

Ces lignes de commandes présupposent que tous les fichiers sont situés dans le même répertoire et que celui-ci est le répertoire courant. Sinon, il faut nommer les fichiers avec leur chemin complet d'accès.

Encadré 4 GET FILE

GET FILE = '*nom_du_fichier.sav*'.

Cette commande demande à SPSS de charger le fichier *nom_du_fichier.sav*.

ATTENTION : si le fichier ne se trouve pas dans le répertoire courant, il faut donner son nom complet précisant le chemin d'accès, somme par exemple 'C:\analyse\spss\travail.sav' .

Encadré 5 COMPUTE

COMPUTE *var = formule*.

Cette commande permet de créer (ou de modifier) la variable *var* en lui affectant les valeurs calculées à partir de *formule*. *formule* peut contenir les opérateurs numériques courants, des fonctions, le nom d'autres variables. Pour plus de renseignements, se reporter à l'aide de SPPS.

3.3 Fusionner les Fichiers.

Nous allons maintenant fusionner tous les fichiers enfant en un seul, le fichier *enfants.sav*. Il est possible de passer par la boîte de dialogue *Données* > *Fusionner des fichiers* > *Ajouter des observations*... . Cette boîte de dialogue permet d'ajouter des observations au fichier actuellement chargé.

Pour cela, il faut ouvrir le fichier *enfant1.sav*, ajouter les observations de *enfant2.sav*, ajouter celles de *enfant3.sav* puis utiliser la commande *Fichier* > *Enregistrer sous*.. pour créer le fichier *enfants.sav*.

Dans le cas présent, passer par l'éditeur de syntaxe est beaucoup plus simple. Il suffit de saisir :

```
ADD FILES
        /FILE='enfant1.sav'
        /FILE='enfant2.sav'
        /FILE='enfant3.sav'.
EXECUTE.

SAVE OUTFILE='enfants.sav'.
```

Encadré 6 ADD FILES

```
ADD FILES
        /FILE='fichier1.sav'
        /FILE='fichier2.sav'
        /FILE='fichier3.sav'
.
EXECUTE.
```

La fonction ADD FILES permet de créer un nouveau fichier de données comportant l'ensemble des observations de *fichier1.sav*, *fichier2.sav* et *fichier3.sav*.
Si des variables ne sont pas communes aux deux fichiers, il est possible de les apparier en les renommant et en leur donnant le même nom.
Les options /DROP, /KEEP et /RENAME sont utilisable avec cette fonction (cf. Encadré 1).

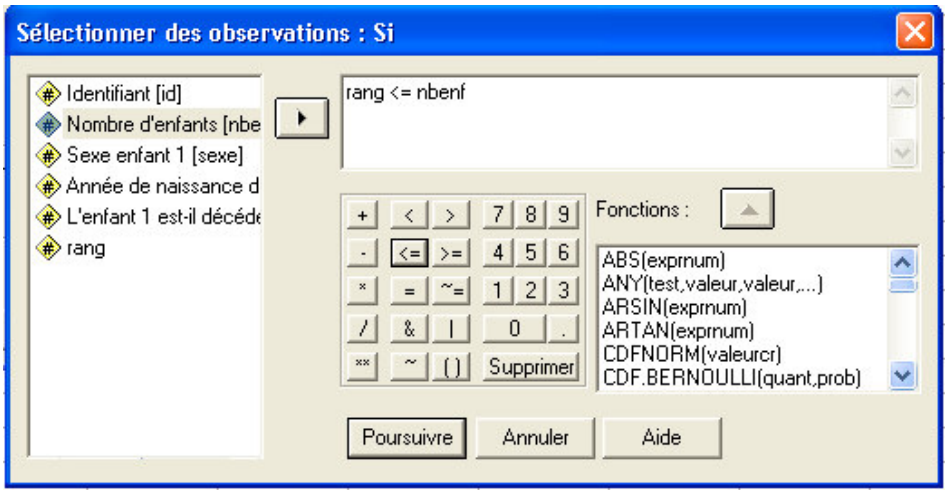
3.4 Suppression des Observations vides.

Il nous faut maintenant supprimer les observations vides résultant par exemple des « enfants » de rang 4 créés pour des personnes ayant moins de quatre enfants.

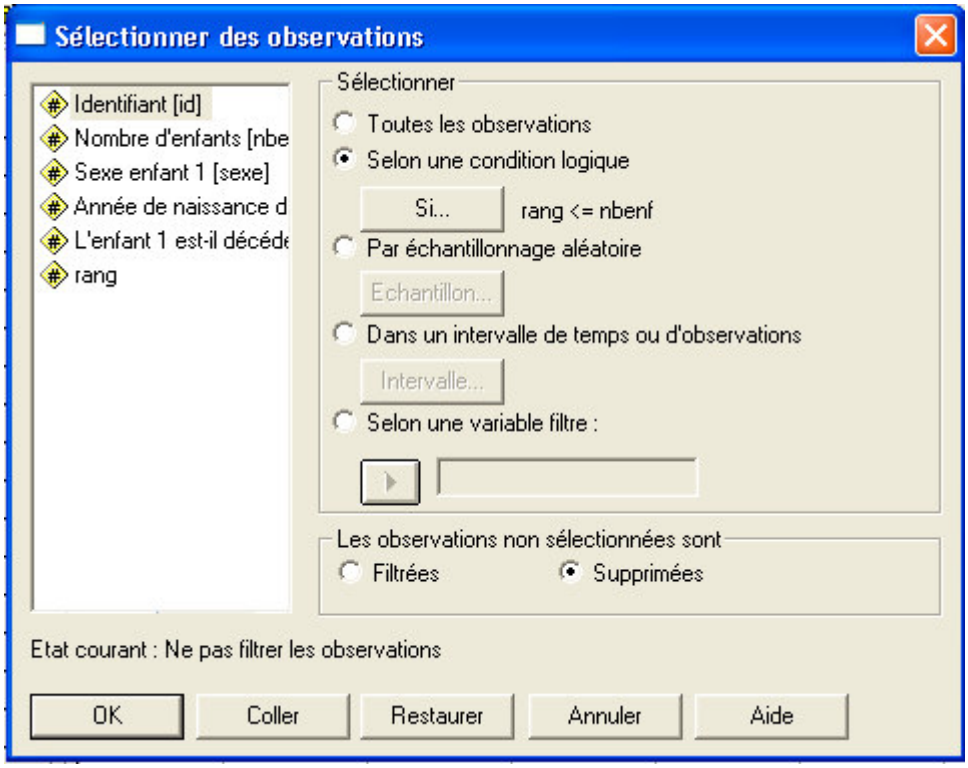
Les observations à supprimer sont celles pour lesquelles *rang* est supérieur à *nbenf*.

Première méthode :

- Cliquer sur *Données* > *Sélectionner des observations*..
- Choisir *Selon une condition logique*.
- Cliquer sur le bouton *Si*..
- Saisir rang<=nbenf.
- Cliquer sur *Poursuivre*.



- Sélectionner *Supprimées*.
- Cliquer sur *OK*.



Seconde méthode :

Saisir dans l'éditeur de syntaxe la commande suivante :

```
SELECT IF(rang <= nbenf).
EXECUTE
```

Encadré 7 SELECT IF

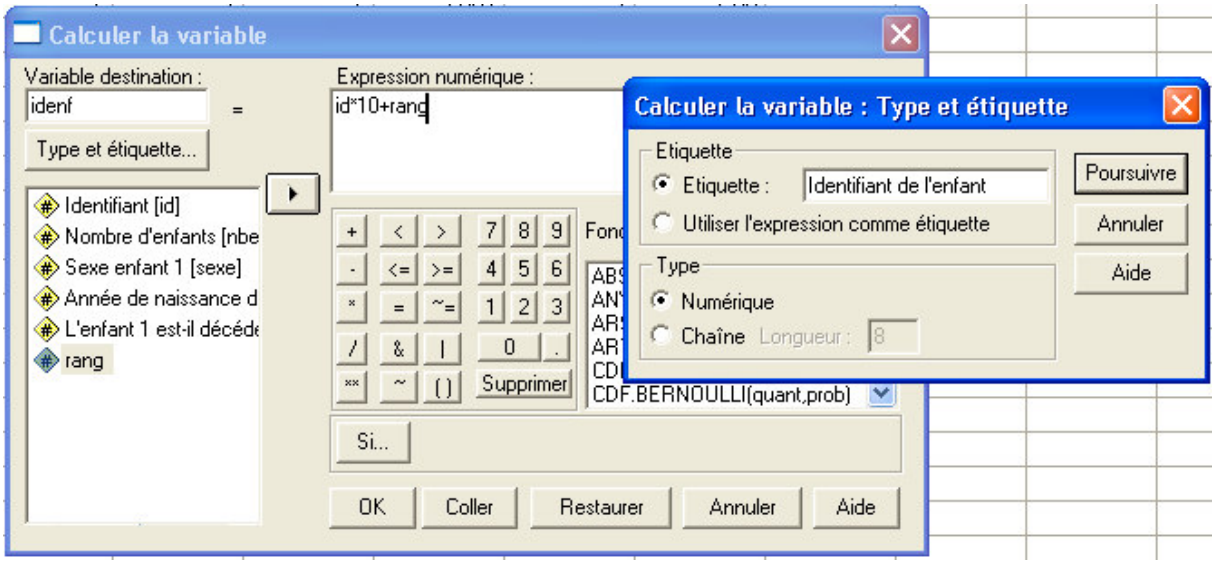
SELECT IF (*condition*).
EXECUTE.

Sélectionne les observations pour lesquelles *condition* est réalisée et supprime les autres.

3.5 Création d'un Identifiant Enfant.

Il est toujours préférable d'avoir un identifiant unique pour chaque observation que nous appellerons *idenf*. Dans le cas présent, le plus simple est de reprendre l'identifiant de la mère et d'y rajouter le rang de l'enfant (ici sur un chiffre car le rang maximum est trois).

Nous pouvons passer par *Tranformer* > *Calculer*... . *idenf* est calculer en faisant l'opération *id**10+*rang*. Il est possible au passage de définit le libellé de la variable *idenf* en cliquant sur *Type et étiquette*... .



Cela se traduit dans l'éditeur de syntaxe par :

```
COMPUTE idenf = id*10+rang .
VARIABLE LABELS      idenf "Identifiant de l'enfant".
EXECUTE .
```

Encadré 8 VARIABLE LABELS

```
VARIABLE LABELS      var1 "Libellé 1"
                     var2 "Libellé 2"
                     var3 "Libellé 3".
```

Cette commande affecte le *libellé1* à la variable *var1*, le *libellé2* à la variable *var2*, etc. On notera que les libellés sont entre crochets.

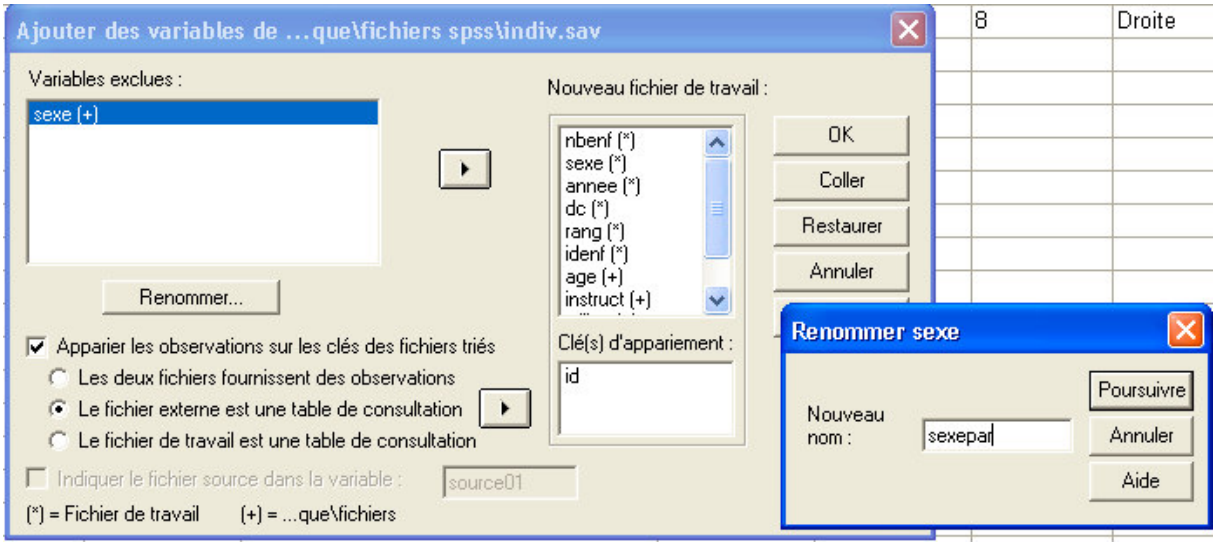
Au passage, on peut en profiter pour corriger les libellés suivants de notre fichier.

```
VARIABLE LABELS      rang "Rang de l'enfant"
                     sexe "Sexe de l'enfant"
                     annee "Année de naissance de l'enfant"
                     dc "L'enfant est-il décéder ?".
```

3.6 Ajouts des Caractéristiques du Parent.

Pour les besoins de l'analyse, on peut avoir besoin de certaines caractéristiques du parent comme son âge, son sexe, etc. Ces différentes caractéristiques sont celles du fichier *indiv.sav*. La variable d'appariement entre *enfants.sav* et *indiv.sav* sera la variable *id*. Le fichier *indiv.sav* sera une table de consultation pour le fichier *enfants.sav*. Ainsi, on affectera à chaque enfant les valeurs de son père ou de sa mère.

- Tout d'abord, il nous faut trier notre fichier selon la variable *id* (cf. 2.1).
- Ensuite, cliquer sur *Données* > *Fusionner des fichiers* > *Ajouter des variables*.. (cf. 2.2).
- Il se trouve que deux variables ont le même nom : *sexe*. Nous allons donc renommer la variable *sexe*, du fichier *indiv.sav*, *sexepar*. Mettre *sexe(+)* en surbrillance.
- Cliquer sur *Renommer*.
- Saisir sexepar.
- Cliquer sur *Poursuivre*.
- Ne pas oublier de passer *sexepar* dans *Nouveau fichier de travail*.
- Sélectionner *Apparier les observations sur les clés des fichiers triés*.
- Sélectionner *Le fichier externe est une table de consultation*.
- Sélectionner *id* comme *Clé(s) d'appariement*.
- Cliquer sur *OK*.



La syntaxe de commande correspondante est :

```
MATCH FILES
    /FILE=*
    /TABLE='indiv.sav'
    /RENAME  sexe=sexepar
    /BY id.
EXECUTE.
```

Encadré 9 MATCH FILES

```
MATCH FILES
        /FILE='fichier_de_travail.sav'
        /TABLE='table_de_consultation.sav'
        /BY clé_d'appariement .
EXECUTE.
```

Cette commande permet d'ajouter des variables issues de *table_de_consultation.sav* à *fichier_de_travail.sav* selon la clé d'appariement *clé_d'appariement*.
Si *fichier_de_travail* est le fichier chargé dans l'éditeur de données, alors remplacer /FILE='*fichier_de_travail.sav*' par /FILE=* .
Il est possible d'utiliser les options /DROP et /RENAME.

3.7 Fichier de Syntaxe.

Bien que l'éditeur de syntaxe, requérant un minimum de programmation, soit d'un usage plus délicat que les boîtes de dialogue, il présente l'avantage de pouvoir sauvegarder les opérations effectuées sous forme d'un fichier de syntaxe. En cas de problème, il est alors possible de refaire toute l'opération en un instant. Il suffit d'ouvrir le fichier de syntaxe, de choisir *Exécuter* > *Tout* pour que le fichier enfants soit de nouveau créer à partir des fichiers sources.

On notera que la majorité des boîtes de dialogue de SPPS dispose du bouton *Coller*. Celui-ci permet, après avoir rempli les différentes valeurs nécessaires dans la boîte de dialogue, de copier dans l'éditeur de syntaxe la syntaxe correspondante aux valeurs saisies dans la boîte de dialogue.

L'ensemble de la création du fichier enfants peut se résumer par le fichier de syntaxe suivant :

```
GET FILE='fecondite.sav'.

SAVE OUTFILE='enfant1.sav'
        /KEEP   id nbenf sexe$1 annee$1
dc$1
        /RENAME
                sexe$1=sexe
                annee$1=annee
                dc$1=dc.
EXECUTE.

SAVE OUTFILE='enfant2.sav'
        /KEEP   id nbenf sexe$2 annee$2
dc$2
        /RENAME
                sexe$2=sexe
                annee$2=annee
                dc$2=dc.
EXECUTE.

SAVE OUTFILE='enfant3.sav'
        /KEEP   id nbenf sexe$3 annee$3
dc$3
        /RENAME
                sexe$3=sexe
                annee$3=annee
                dc$3=dc.
EXECUTE.

GET FILE='enfant1.sav'.
COMPUTE rang = 1 .
EXECUTE .
SAVE OUTFILE='enfant1.sav'.

GET FILE='enfant2.sav'.
COMPUTE rang = 2 .
EXECUTE .
SAVE OUTFILE='enfant2.sav'.

GET FILE='enfant3.sav'.
COMPUTE rang = 3 .
EXECUTE .
SAVE OUTFILE='enfant3.sav'.

ADD FILES
        /FILE='enfant1.sav'
        /FILE='enfant2.sav'
        /FILE='enfant3.sav'.
EXECUTE.

SAVE OUTFILE='enfants.sav'.

SELECT IF(rang <= nbenf).
EXECUTE .

COMPUTE idenf = id*10+rang .
VARIABLE LABELS
        idenf "Identifiant de l'enfant"
        rang "Rang de l'enfant"
        sexe "Sexe de l'enfant"
        annee "Année de naissance de
l'enfant"
        dc "L'enfant est-il décéder ?".
EXECUTE .

GET FILE='indiv.sav'.
SORT CASES BY id (A) .
EXECUTE.
SAVE OUTFILE='indiv.sav'.

GET FILE='enfants.sav'.
SORT CASES BY id (A) .
EXECUTE.
SAVE OUTFILE='enfants.sav'.

MATCH FILES
 /FILE=*
 /TABLE='indiv.sav'
 /RENAME  sexe=sexepar
 /BY id.
EXECUTE.

SAVE OUTFILE='enfants.sav'.
```