

La proc TRANSPOSE expliquée à ma fille

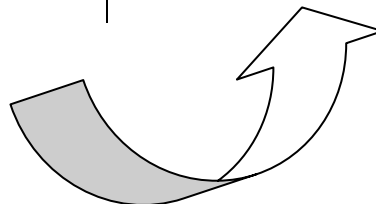
De nombreux cas de figure dans SAS nécessitent de réorganiser les données de manière à changer ce que représente une observation dans une table. Des pivots (en général partiels) dans les données sont alors à envisager ; la procédure TRANSPOSE est l'outil privilégié pour les pivots.

C'est quoi un pivot ?

L'exemple ci-dessous illustre un pivot partiel de la variable Y selon la variable IDENT. La variable X, elle, est perdue. A l'intérieur de chaque bloc de IDENT dans la table d'origine, chaque observation vient alimenter une colonne dans la table transposée.

IDENT	X	Y
1	a	15
1	z	3
1	w	-1
3	a	1
4	w	0
4	z	.
4	b	15
4	a	3

IDENT	VAR1	VAR2	VAR3	VAR4
1	15	3	-1	.
3	1	.	.	.
4	0	.	15	3

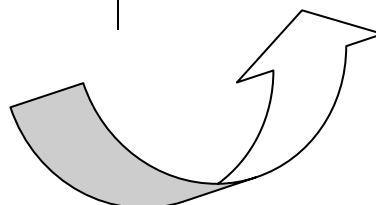


On peut également utiliser les valeurs d'une variable pour nommer les nouvelles variables, plutôt que d'utiliser un nommage automatique comme ci-dessous (VAR1, VAR2, VAR3 sont des noms automatiques).

Voici un second pivot partiel de la même table. Cette fois, on utilise les valeurs de la variable X comme nom des nouvelles variables. Attention, ces valeurs doivent être des noms de variables SAS licites ; de plus, on ne doit pas avoir de doublons de X à l'intérieur d'un bloc de ID.

IDENT	X	Y
1	a	15
1	z	3
1	w	-1
3	a	1
4	w	0
4	z	.
4	b	15
4	a	3

IDENT	a	z	w	b
1	15	3	-1	.
3	1	.	.	.
4	3	.	0	15



Syntaxe simplifiée de la procédure Transpose

```
PROC TRANSPOSE DATA = tableLue OUT = tableTransposée PREFIX = ... ;
  BY ... ;
  VAR ... ;
  ID ... ;
RUN ;
```

On donne dans PREFIX la racine commune des noms de nouvelles variables, en cas de nommage automatique. Dans BY, on indique la variable fixe du pivot (dans les exemples précédents, la variable IDENT) ; dans VAR, la variable à transposer (Y dans l'exemple) ; dans ID, une variable contenant le nom des nouvelles colonnes à alimenter (X dans l'exemple).

Cas 1 : transposer une variable (facile !)

```
PROC SORT DATA = sashelp.class OUT = work.class ;
  BY sex ;
RUN ;
PROC TRANSPOSE DATA = work.class OUT = work.Tclass ;
  BY sex ;
  VAR age ;
RUN ;
```

Sex	Age
M	14
F	13
F	13
F	14
M	14
M	12
F	12
F	15
M	13
M	12
F	11
F	14
F	12
F	15
M	16
M	12
M	15

Avant

Sex	_NAME_	COL1	COL2	COL3	COL4	COL5	COL6	COL7	COL8	COL9	COL10
F	Age	13	13	14	12	15	11	14	12	15	.
M	Age	14	14	12	13	12	16	12	15	11	15

Après

Cas 2 : transposer plusieurs variables (moins immédiat, 2 solutions)

Solution 1 : faire une proc Transpose par variable, puis fusionner (MERGE) les tables transposées en une seule

```
PROC SORT DATA = sashelp.class OUT = work.class ;
  BY sex ;
RUN ;
PROC TRANSPOSE DATA = work.class OUT = work.Tage PREFIX=age ;
  BY sex ;
  VAR age ;
RUN ;
```

```

PROC TRANSPOSE DATA = work.class OUT = work.Tnom PREFIX=nom ;
  BY sex ;
  VAR name ;
RUN ;
DATA work.Tclass ;
  MERGE work.Tage work.Tnom ;
  BY sex ;
RUN ;

```

Sex	_NAME_	age1	age2	age3	age4	age5	age6	age7	age8	age9	age10	nom1	nom2
F	Name	13	13	14	12	15	11	14	12	15	.	Alice	Barbara
M	Name	14	14	12	13	12	16	12	15	11	15	Alfred	Henry

nom3	nom4	nom5	nom6	nom7	nom8	nom9	nom10
Carol	Jane	Janet	Joyce	Judy	Louise	Mary	
James	Jeffrey	John	Philip	Robert	Ronald	Thomas	William

Solution 2 : préparer les données en doublonnant les lignes avec une étape Data. Convient mieux pour transposer un grand nombre de variables.

```

PROC SORT DATA = sashelp.class OUT = work.class ;
  BY sex ;
RUN ;
DATA work.class ;
  SET work.class ;
  BY sex ;
  IF FIRST.sex THEN compteur = 0 ;
  compteur + 1 ;
  nomVar = COMPRESS("nom"!!compteur) ;
  valeur = name ; /* pas de PUT car données caractère */
  OUTPUT ; /* écrit une ligne */
  nomVar = COMPRESS("age"!!compteur) ;
  valeur = PUT(age, 3.) ; /* PUT car données numériques */
  OUTPUT ;
RUN ;
PROC TRANSPOSE DATA = work.class OUT = work.Tclass ;
  BY sex ;
  VAR valeur ;
  ID nomVar ;
RUN ;

```

Sex	_NAME_	nom1	age1	nom2	age2	nom3	age3	nom4	age4	nom5	age5	nom6	age6
F	valeur	Alice	13	Barbara	13	Carol	14	Jane	12	Janet	15	Joyce	11
M	valeur	Alfred	14	Henry	14	James	12	Jeffrey	13	John	12	Philip	16

nom7	age7	nom8	age8	nom9	age9	nom10	age10
Judy	14	Louise	12	Mary	15		
Robert	12	Ronald	15	Thomas	11	William	15