

La proc GPLOT expliquée à ma fille

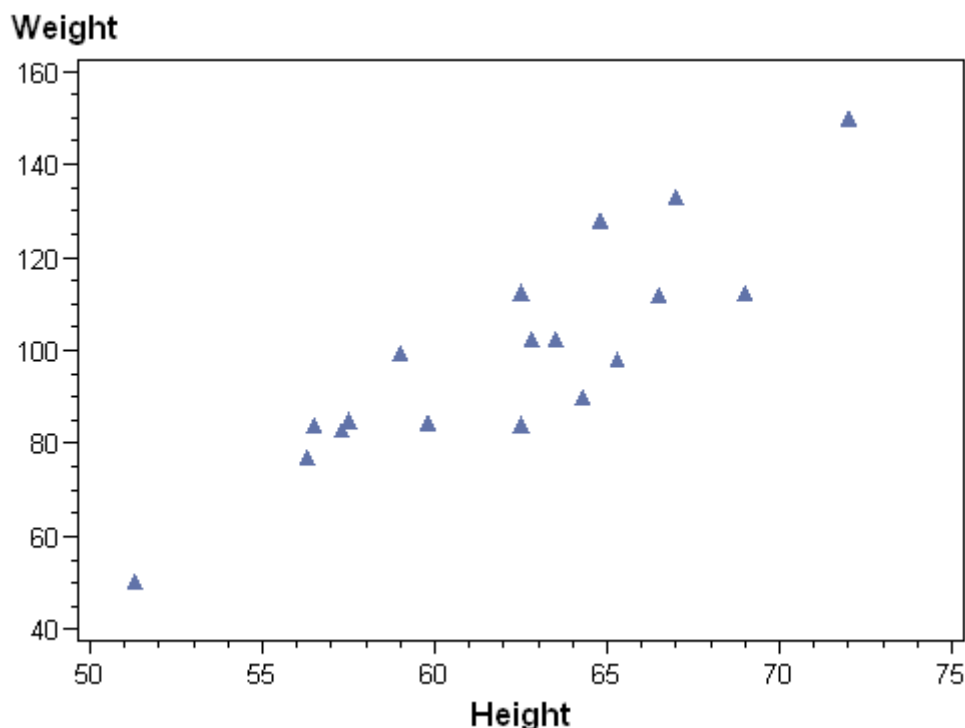
La production de graphiques depuis SAS a mauvaise presse. Pourtant sa syntaxe n'est pas si compliquée qu'on le dit, et son rendu graphique est largement acceptable, surtout si on ne joue pas au jeu des 10 erreurs avec un graphique sous Excel.

Le cœur de chauffe : la procédure elle-même

```
PROC GPLOT DATA = tableLue ;  
    PLOT variableAxeVertical * variableAxeHorizontal ;  
RUN ; QUIT ;
```

A la lecture de cette syntaxe : deux questions viennent souvent à l'esprit. 1) C'est juste ça ? Pas la peine d'en faire toute une histoire alors ! et 2) pourquoi un QUIT à la fin en plus du RUN ? Pour des raisons qui remontent à la préhistoire de SAS, qui n'ont plus aucun intérêt pratique aujourd'hui, mais si on ne met pas ce QUIT, la procédure GPLOT ne sera pas considérée comme terminée (le temps d'exécution affiché dans la Log sera donc décompté, et les fenêtres de SAS auront un « Proc GPLOT running » dans leur bandeau) jusqu'à la prochaine procédure ou étape Data qui sera exécutée.

```
PROC GPLOT DATA = sashelp.class ;  
    PLOT weight * height ;  
RUN ; QUIT ;
```



L'instruction SYMBOL

Par défaut, cette procédure GPLOT sans plus de précisions produit un nuage de points. Pour en faire des courbes, on ne doit rien ajouter à la procédure elle-même, mais mettre dans le

programme des instructions périphériques, qui sont comme les titres et les pieds de page à effet permanent tout au long de la session SAS. La principale instruction associée à GPLOT est SYMBOL : elle permet de définir les marqueurs des points.

```
SYMBOL i v = typeMarqueur i = typeGraphique c = couleur ;
```

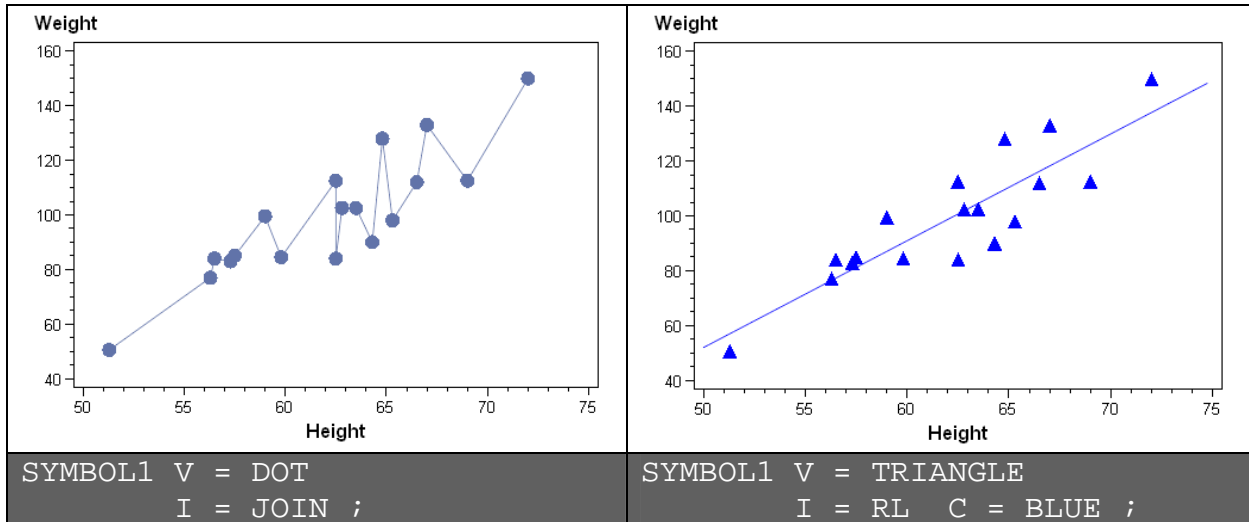
Comme les titres, les instructions SYMBOL sont numérotées. Il peut y en avoir jusqu'à 99. Les options sont :

- le type de marqueur (PLUS, DOT, CIRCLE, SQUARE, DIAMOND, TRIANGLE, ...) qui est le caractère utilisé pour repérer les points. V=NONE permet de ne pas marquer les points ;

PLUS	+	%	(percent)	♣
X	×	&	(ampersand)	♠
STAR	*	'	(single quote)	♣
SQUARE	□	=	(equals)	☆
DIAMOND	◇	-	(hyphen)	⊙
TRIANGLE	△	@	(at)	♀
HASH	#	*	(asterisk)	♀
Y	Y	+	(plus)	⊕
Z	Z	>	(greater than)	♂
PAW	⋯	.	(period)	℥
POINT	.	<	(less than)	℥
DOT	●	,	(comma)	♁
CIRCLE	○	/	(slash)	♁
_ (underscore)	□	?	(question mark)	ℙ
" (double quote)	♠	((left parenthesis)	☾
# (pound sign)	♥)	(right parenthesis)	♁
\$ (dollar sign)	◇	:	(colon)	✱

- le type de graphique : I=NONE construit un nuage de points, I=JOIN relie tous les points (attention : **dans l'ordre où ils sont dans la table SAS**), I=SPLINE fait une courbe lissée et I=RL superpose une droite de régression linéaire au nuage de points (l'équation de régression est donnée dans la Log) ;
- la couleur des marqueurs et des courbes éventuelles (voir « les couleurs de SAS expliquées à ma fille »).

Attention, certains drivers graphiques comme Java ou ActiveX ne respectent pas toujours les injonctions de l'instruction SYMBOL.

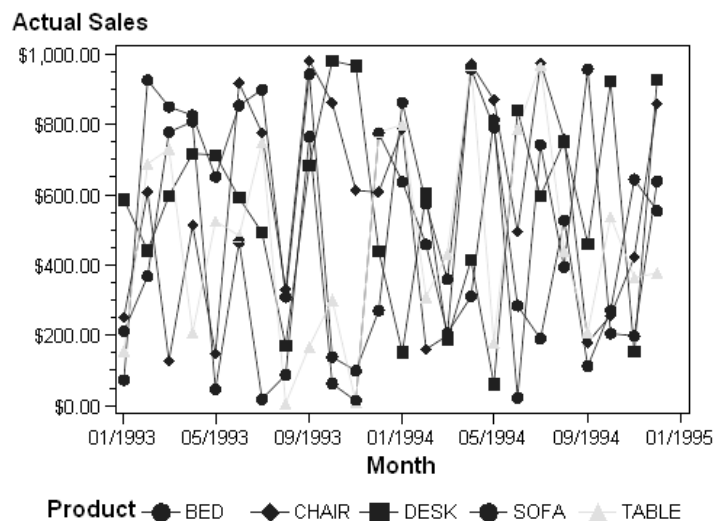


Plusieurs séries de points

On peut utiliser deux formes d'instruction PLOT pour la superposition de plusieurs séries de points sur un seul graphique. La disposition des données dans la table SAS lue dicte le choix entre les deux écritures.

<p>Cas ❶ : Les séries à distinguer se trouvent dans deux variables séparées. L'option OVERLAY permet de superposer les courbes (sinon, SAS édite 2 graphiques distincts).</p>	<p>Cas ❷ : Les séries à distinguer se trouvent à des observations séparées. Une variable de groupe permet de distinguer, par ses valeurs, les différentes séries.</p>
<pre>PLOT (varY1 varY2) * varX / LEGEND OVERLAY ;</pre>	<pre>PLOT varY * varX = varGroupe ;</pre>

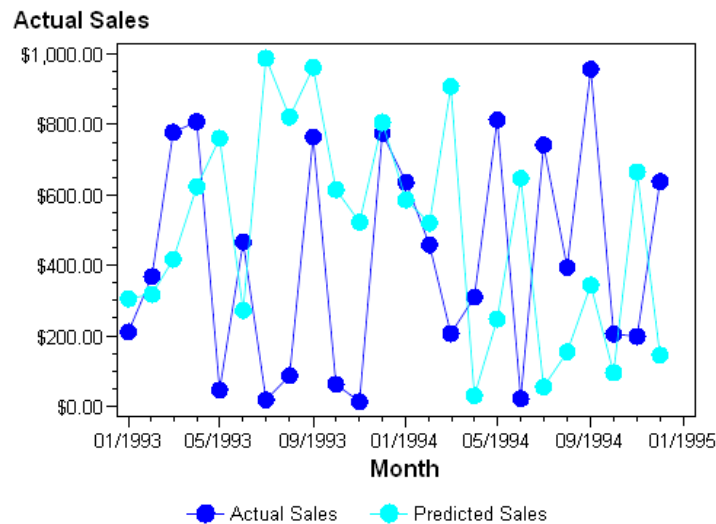
```
SYMBOL1 i = join c = gray20 v = circle ;
...
SYMBOL5 i = join c = grayE0 v = triangle ;
PROC GPLOT DATA = work.sales ;
  PLOT actual * month = product ;
  FORMAT month MMYYS7. ;
RUN ; QUIT ;
```



```

SYMBOL1 i = join c = blue v = circle ;
SYMBOL2 i = join c = cyan v = dot ;
PROC GPLOT DATA = work.sales (WHERE = (product = "BED")) ;
  PLOT (actual predict) * month / OVERLAY LEGEND ;
  FORMAT month MMYYS7. ;
RUN ; QUIT ;

```



Une troisième syntaxe permet, avec une instruction PLOT2 supplémentaire, de construire des graphiques à double échelle.

```

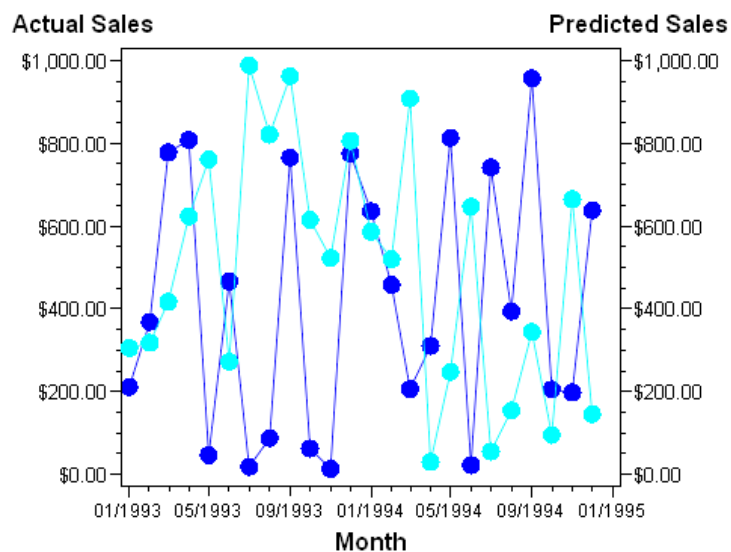
PLOT variableAxeVertical1 * variableAxeHorizontal ;
PLOT2 variableAxeVertical2 * variableAxeHorizontal ;

```

```

PROC GPLOT DATA = work.sales (WHERE = (product = "BED")) ;
  PLOT actual * month / LEGEND ;
  PLOT2 predict * month / LEGEND ;
  FORMAT month MMYYS7. ;
RUN ; QUIT ;

```



Graphiques en bulles

Ce type de graphique est une variante du nuage de points ; les marqueurs y sont proportionnels à la valeur d'une troisième variable. L'instruction n'est plus PLOT mais BUBBLE. Pour les graphiques à double échelle, il existe une instruction BUBBLE2 identique dans son principe à PLOT2. L'option BLABEL affiche la valeur de la variable de taille à côté de chaque point. Si un format est associé à cette variable, les valeurs affichées seront formatées mais ce seront les valeurs non formatées qui auront servi à calculer la taille des points.

```
PROC GPLOT DATA = tableLue ;  
    BUBBLE varY * varX = varTaille / BLABEL ;  
RUN ; QUIT ;
```

```
PROC SUMMARY DATA = sashelp.shoes NWAY ;  
    VAR sales ;  
    CLASS product ;  
    OUTPUT OUT = work.sales MEAN= ;  
RUN ;  
PROC GPLOT DATA = work.sales ;  
    BUBBLE sales * product = _freq_ / BLABEL ;  
RUN ; QUIT ;
```

Total Sales

