

Modifying Graphics in SAS

Statistics 135

Autumn 2005



Modifying Graphs

As in **S**, it is possible to modify fonts, colours, symbols, lines, etc in **SAS**. The approach is a bit different, as much of these changes are done outside the PROC used for graphing.

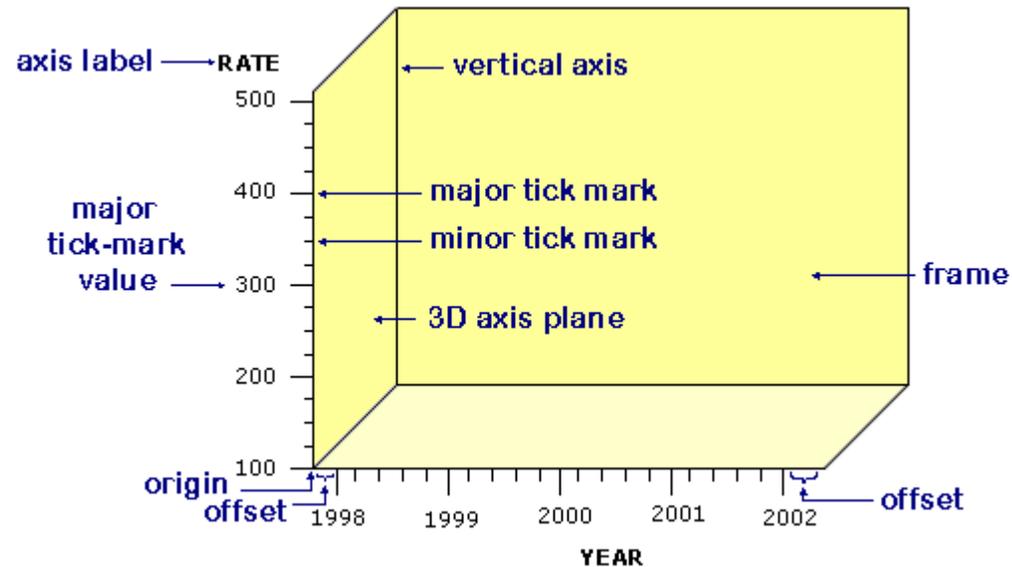
These can be controlled with the following statements (which in many cases should be placed just before the PROC they are to be applied to)

- **AXIS**: Control axes in GBARLINE, GCHART, GCONTOUR, GPLOT, and GRADAR procedures.

The basic structure in

```
AXIS<1:99> <OPTIONS> ;
```

You must give a number when defining an axis. This also allows for an axis definition to be used in multiple plots.



The important options are

- LABEL: Axis label. Like xlab and ylab in **S**.
- ORDER: Positions of major tick marks.
- VALUE: Text for major tick marks.
- COLOR: Colour of axis.
- LOGBASE = base | E | PI: Allows for logarithmic axis scale.
- LOGSTYLE = EXPAND | POWER: EXPAND shows actual values while POWER shows log values.

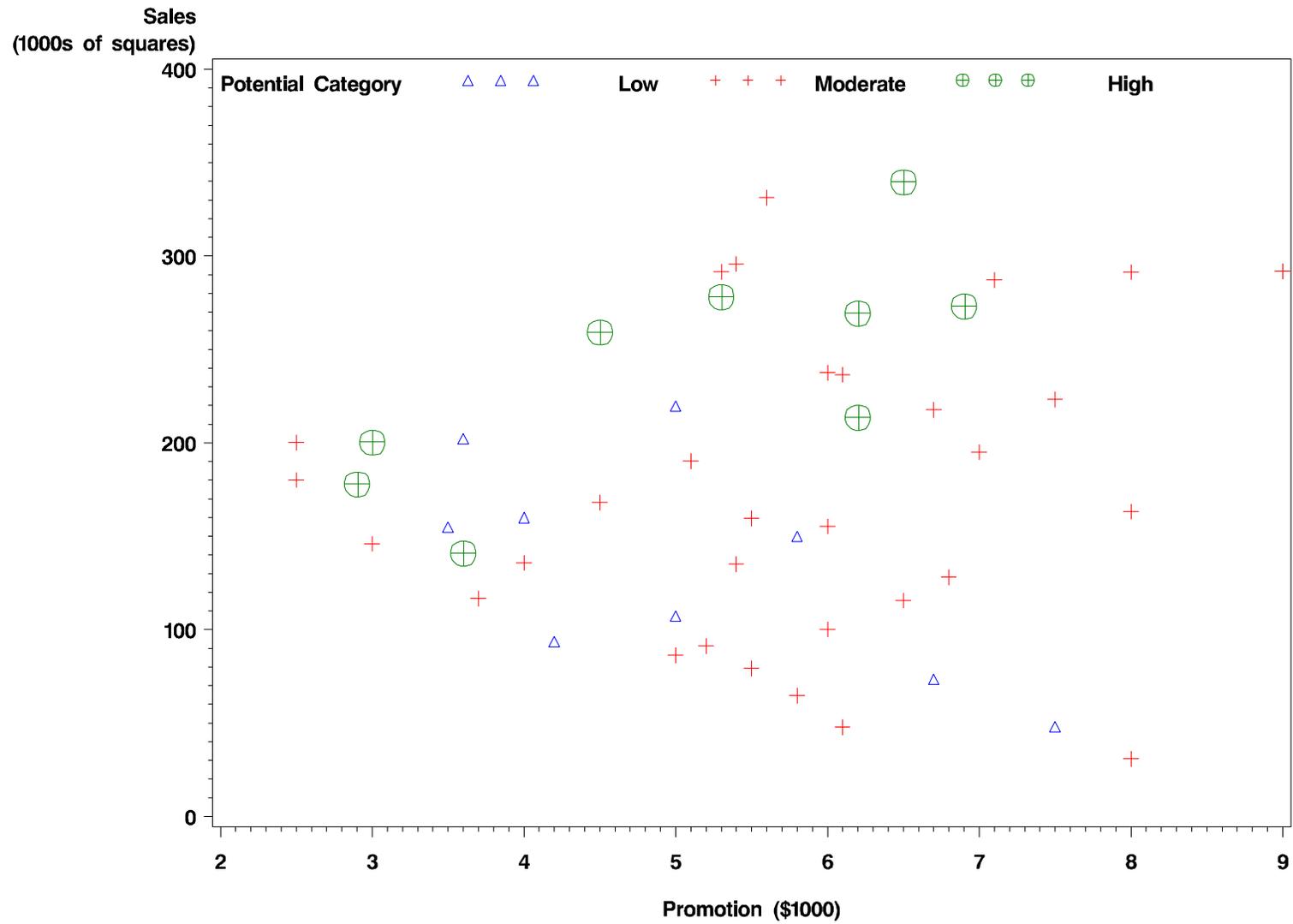
- MAJOR: Configures major tick marks.
- MINOR: Configures major tick marks.

Each of these options may take multiple suboptions.

For example, let look at

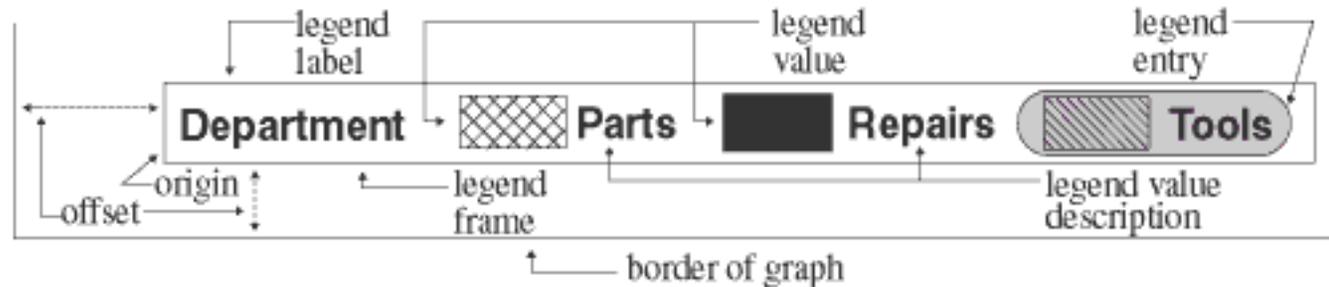
```
AXIS1 LABEL = ("Promotion ($1000)");  
AXIS3 LABEL = (COLOR = cyan 'Sales'  
               justify=right '(1000s of squares)' )  
               COLOR = green  
               MINOR = (NUMBER = 3 COLOR = red);  
PROC GPLOT DATA=shingles2;  
  PLOT sales*promotion / HAXIS=axis1  
                       VAXIS=axis3;
```

Sales vs Promotion



Shingle Sales Data

- LEGEND: Controls the legend in GCHART, GCONTOUR, GMAP, and GPLOT procedures.



Important options include

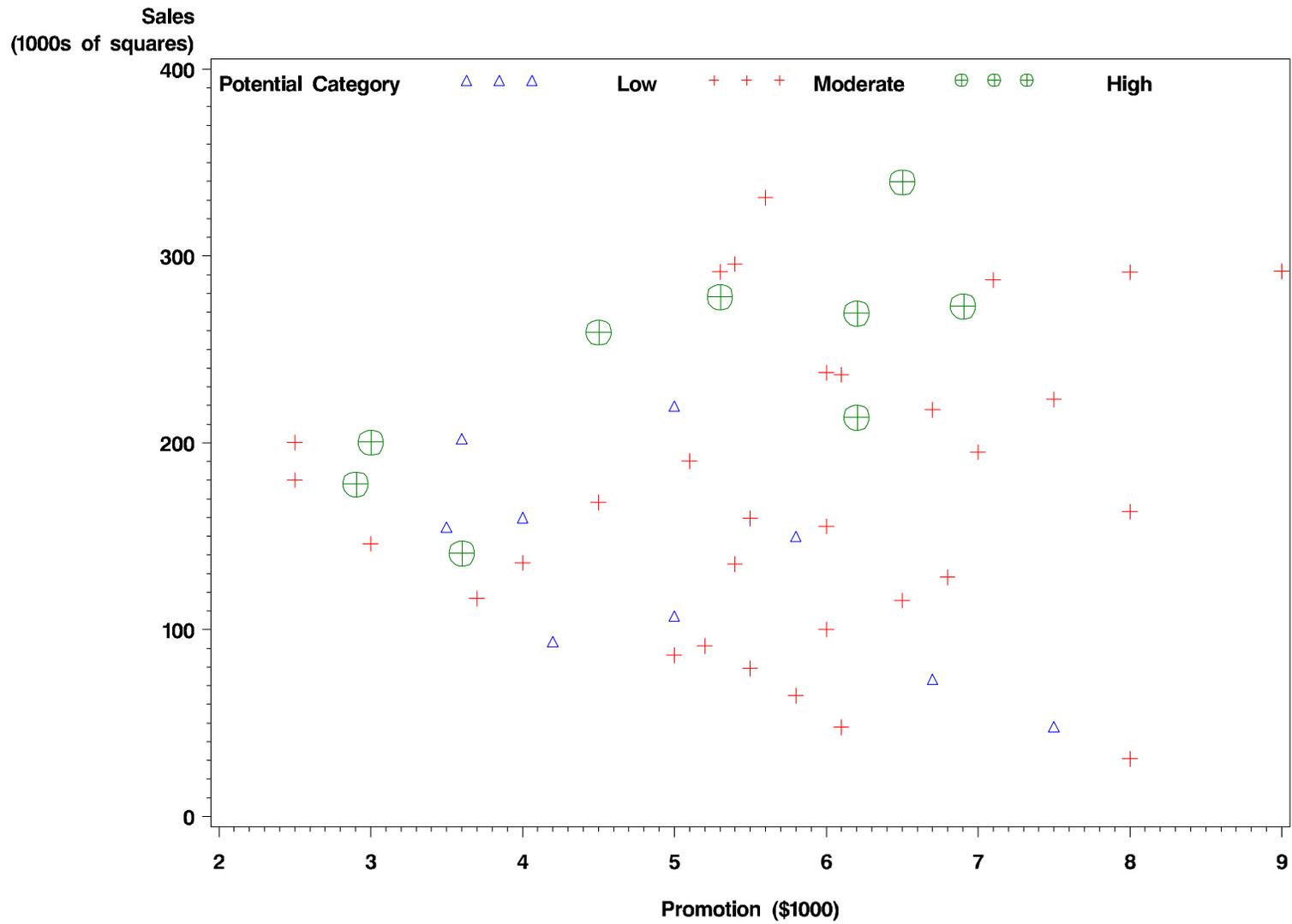
- LABEL: A label for the legend
- VALUE: Description of each level
- ORDER: Order that values appear in the legend
- POSITION=(`<BOTTOM | MIDDLE | TOP> <LEFT | CENTER | RIGHT> <INSIDE | OUTSIDE>`): Where to position the legend
- ACROSS=`n` Number of columns
- DOWN=`n` Number of rows

Here is an example of a legend

```
LEGEND1 LABEL = ('Potential Category')
          VALUE = ('Low' 'Moderate' 'High')
          POSITION = (top left inside);

PROC GGPLOT DATA=shingles2;
  PLOT sales*promotion=pot / HAXIS=axis1
                             VAXIS=axis2
                             LEGEND = LEGEND1;
```

Sales vs Promotion



Shingle Sales Data

- PATTERN: Controls the use of patterns in graphs created by GCHART, GCONTOUR, GMAP, GPLOT procedures. Also used as part of the SYMBOL statement.

The structure of the statement looks like

```
PATTERN<1...255>  
<COLOR=pattern-color>  
<REPEAT=number-of-times>  
<VALUE=bar/block-pattern  
    | map/plot-pattern  
    | pie/star-pattern  
    | hardware-pattern>;
```

For example

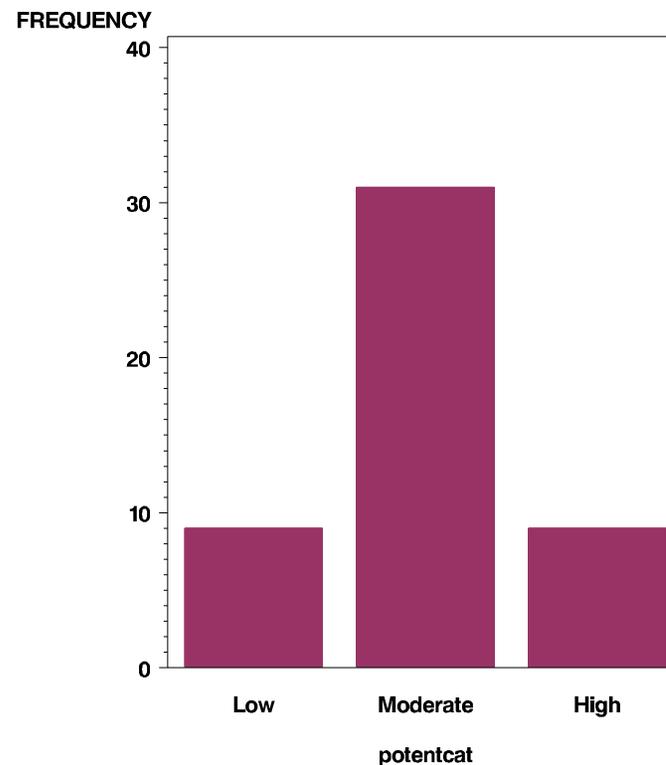
```
PATTERN1 COLOR=CX993366 ;
```

```
PROC GCHART DATA = shingles2;
```

```
  TITLE 'Breakdown of Region Potential Classes';
```

```
  VBAR potentcat / MIDPOINTS = 'Low' 'Moderate' 'High';
```

Breakdown of Region Potential Classes



- SYMBOL: Controls symbols in GBARLINE, GCHART, GCONTOUR, GPLOT, and GRADAR procedures.

The structure of the statement is

```
SYMBOL<1...255>  
  <COLOR=symbol-color>  
  <MODE=EXCLUDE | INCLUDE>  
  <REPEAT=number-of-times>  
  <STEP=distance<units>>  
  <appearance-option(s)>  
  <interpolation-option>  
  <SINGULAR=n>;
```

Important options include

– VALUE: Indicates what symbol to use. From the documentation page

VALUE=	Plot Symbol	VALUE=	Plot Symbol
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)	ψ
_ (underscore)	□	? (question mark)	ℙ
" (double quote)	♠	((left parenthesis)	☾
# (pound sign)	♥) (right parenthesis)	♂
\$ (dollar sign)	◇	: (colon)	*

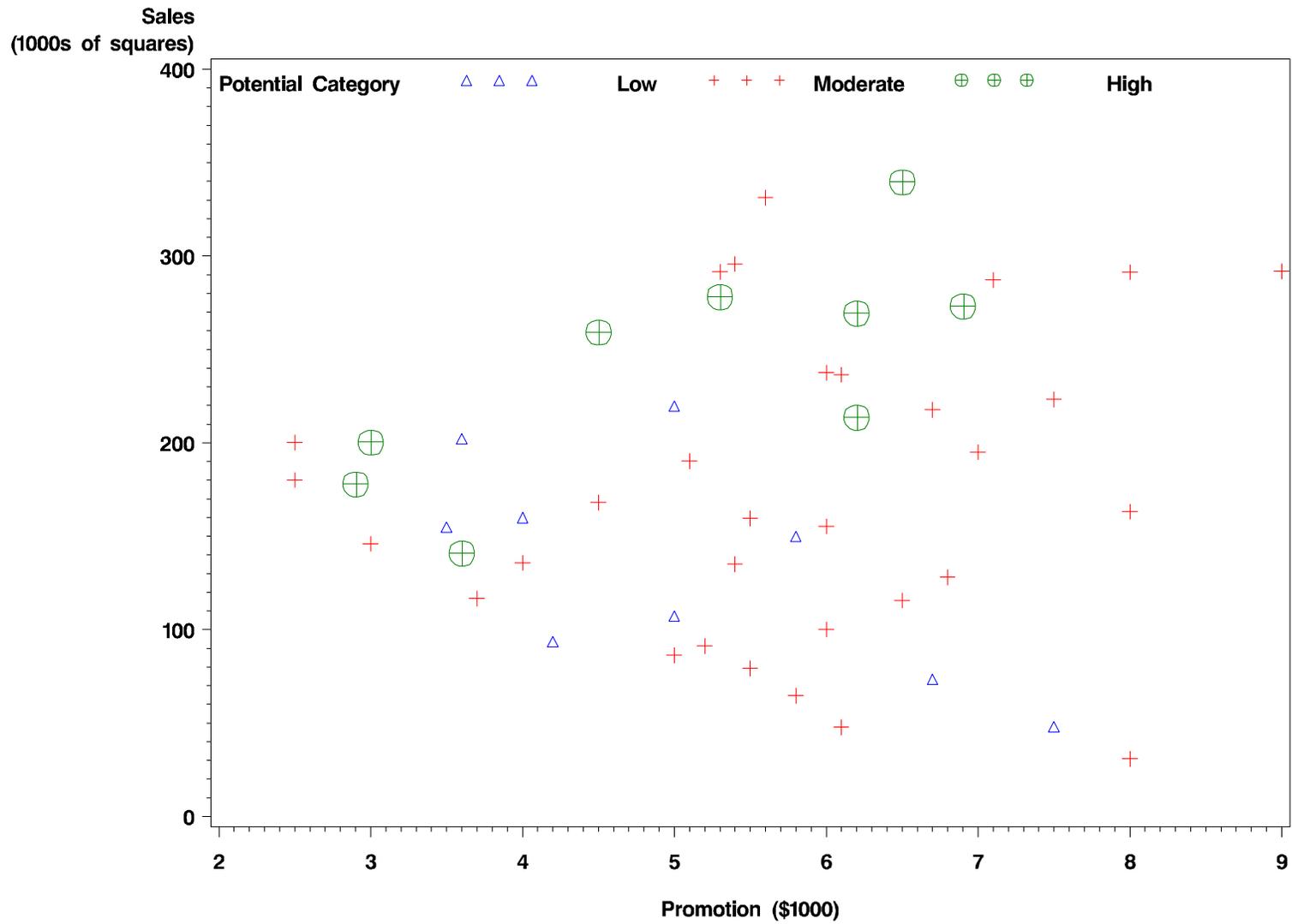
- HEIGHT: Indicates the size of the symbol. In the default setup, it acts like a multiplier.
- INTERPOL: Indicates how points should be joined, if at all. Some of the possibilities include
 - * JOIN: Join points by straight lines. Like `type="1"` in **S**
 - * BOX: Box plot for each X
 - * HILO: Vertical lines between min and max Y values for each X
 - * L: Lagrangian interpolation polynomial
 - * NEEDLE: Draw vertical line from Y to 0 for each point
 - * R: Regression fits. Can handle linear (L), quadratic (Q) and cubic (C) fits. Can add confidence intervals for fits and prediction intervals.
 - * STEP: Step plot
- LINE: Line type. There are 46 of them

Here is an example of SYMBOL statements. In this example, a SYMBOL statement is needed for each level of pot, which is the reason for three of them. If there is only one level in the plot only one is needed.

```
SYMBOL1 COLOR=blue VALUE=triangle height=1;  
SYMBOL2 COLOR=CXFF0000 VALUE=plus height=1.5;  
SYMBOL3 COLOR=green VALUE=+ height=2;
```

```
PROC Gplot DATA=shingles2;  
  PLOT sales*promotion=pot / HAXIS=axis1  
                           VAXIS=axis2  
                           LEGEND = LEGEND1;
```

Sales vs Promotion



Shingle Sales Data

- **FOOTNOTE:** Adds footnotes to plots in many PROCs. Works similarly to adding footnotes to regular output.
- **TITLE:** Affects titles on plots, similarly to the use of the command in general output.

Both of these take the same options. The important ones of these include

- **COLOR:** Font colour. There is a number of ways to indicate fonts. This can be done by name, RGB codes (which look like CRRGGGBB), CYMK codes, plus others as well.
- **FONT:** Type face. Check the documentation for the possibilities, which include CENTURY, SWISS, GERMAN, HERSHEY, MARKER (useful for plotting symbols). Adding B at the end of the name will bold the font, I will italicize it, and BI will bold-italicize it (in most cases).
- **JUSTIFY = LEFT | CENTER | RIGHT:** Justification of text.

Like TITLES and FOOTNOTES in regular output, you can use number to have multiple lines.

GOPTIONS

GOPTIONS is used to setup graphics options in **SAS**. These can either be global options or ones specific to a single plot. These can be used for setting fonts FTEXT, FTITLE, HTEXT, HTITLE, colour COLOR, CBACK, CBY, CPATTERN, CSYMBOL, and how output is sent to output. GOPTIONS is the **SAS** equivalent to par is **S**.

One important options is

```
RESET = ALL
```

which does what it says. Generally, changing a setting will make it effective for all following plots, unless they are reset

Writing Figures to Files

As mentioned before, **SAS** graphics can be written out in many format. While some can be done by choosing Save from the File menu, often it is better to save the file with **SAS** code.

To save your file in a specific format you need to know the appropriate device driver. This can be determined by the code

```
proc gdevice catalog=sashelp.devices;
```

However ones you might find useful are

Device Driver	Format
pslepsc	Colour Encapsulated Postscript (*)
psepsf	Gray-scale Encapsulated Postscript (*)
pscolor	Colour Postscript
ps	Gray-scale Postscript
pdf	Portable Document Format (*)
emf	Enhanced Windows Metafile (*)
wmf	Windows Metafile
gif	Graphics Interchange Format (*)
tiff	TIFF

The (*) ones are probably most useful. Encapsulated postscript for \LaTeX , emf for Windows applications like Word, and gif for web uses. For gif files it is possible to indicate the size of the image with a more precise driver specification. To get the largest sized gif from **SAS**, use gif733 (I think).

To store a figure via **SAS** code, you will need code along the lines of

```
FILENAME file-reference 'file-location';  
GOPTIONS DEVICE = desired device  
          GSFNAME = file-reference  
          GSFMODE = REPLACE | APPEND
```

Plot commands

You might need some additional options, such as to specify the size of a postscript image. The GSFMODE option only will work with some graphic formats as appending pages to a file will only work with a limited number of formats.

To save one the graphs used I place the following code just before the plotting code.

```
FILENAME grafout5 "F:\SAS\salespromo2.eps";
GOPTIONS device=pslpsfc
          GSFNAME=grafout5
          GSFMODE=REPLACE
          HSIZE = 7IN
          VSIZE = 5IN;
```

For each file name

This will create a colour, encapsulated postscript file 7 inches wide by 5 inches tall.