Paper 2242 -2015

CREATIVE USES OF VECTOR PLOTS USING SAS®

Deli Wang, Ossining, NY; Chunxia Lin, Ossining, NY

ABSTRACT

Hysteresis loops occur because of a time lag between input and output. In pharmaceutical industry, hysteresis plots are tools to visualize the time lag between drug concentration and drug effects. Before SAS 9.2, SAS annotations can be used to generate such plots. One of the criticisms is that SAS programmers have to write complex macros to automate the process. Code management and validation tasks are not easy. With SAS 9.2, SAS programmers are able to generate such plots with ease. This paper demonstrates the generation of such plots with both SAS annotation and SAS Graph 9.2.

INTRODUCTION

From SAS 9.2, SAS has made a significant progress in graphing procedures. One of the most significant changes is the release of statistical graphing tools like PROC SGPLOT and SAS Graphic Template Language (GTL). Both provided a lot of new statements. SAS programmers can now tackle many hard plotting tasks with easy codes: overlaying plots of different type, plotting multiple panels, putting error bars and so on so force. Generating hysteresis plots with new procedure is the one that can really make you "WOW".

HYSTERESIS PLOTS: FIRST LOOK

Figure 1 is an example of a hysteresis plot. The data points (X0 and Y0) were collected after taking drug x. The mean data readings for treatment 1 at Day 1, 15, 29, 36, 43, 57, 71 and 85 are calculated below (**Table 1**). Hysteresis plots were plotted by "Y0" against "X0" instead of plotting "Y0" or "X0" against "Day", also data points on hysteresis plots were connected by lines (usually arrows) in the order of time sequence. The drug effects on "X0" and "Y0" were not occurred at the same rate, so a loop occurred. The loop will be bigger when there is a more significant time lag between Y0 and X0. Such plots are not popular, but they are good to examine the effects with time in PD analysis.

Table 1:

TRT	X0	Y0	DAY	N	SE
1	0.50	13.60	1	2	0.5
1	8.81	13.15	15	2	0.5
1	45.20	8.95	29	2	0.5
1	34.20	5.60	36	2	0.5
1	25.00	6.40	43	2	0.5
1	20.60	6.50	57	2	0.5
1	0.50	8.35	71	2	0.5
1	0.50	9.55	85	2	0.5

Hysteresis Plot

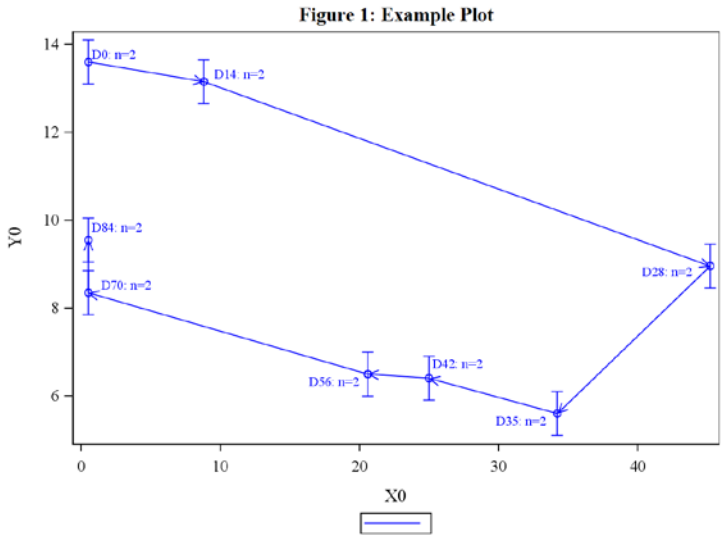


Figure 1. Hysteresis Plot

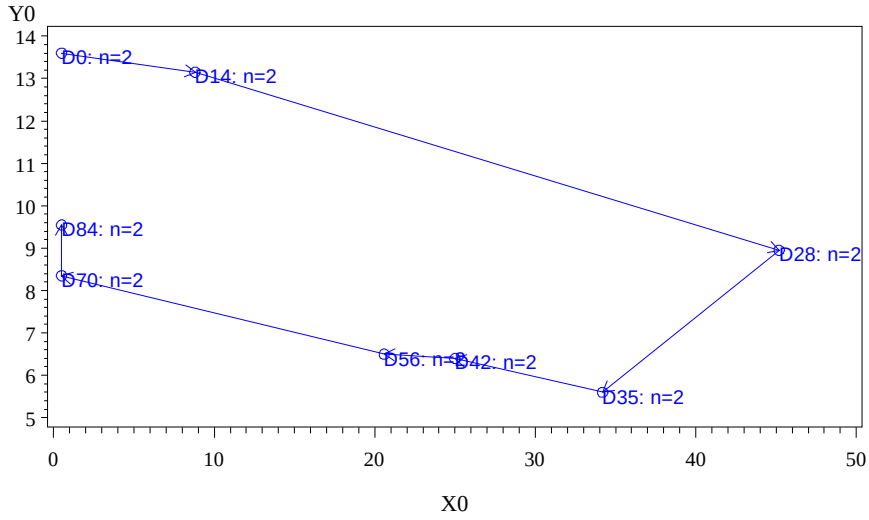
PLOTS WITH SAS ANNOTAIONS

The sample data were manipulated to get the lagged value of both X0 and Y0 to pair with the time points, so that the paired points could be connected with different lines or arrows.

The following code illustrated how to plot a hysteresis plot using SAS annotation.

```
goptions reset=all  center;

DATA W;
   INPUT TRT  X0  Y0  DAY   N  SE;
              D='D'||STRIP(day-1);
              label=catx(': n=',D,n);
              XLL=lag(X0 );
              YLL=lag(Y0);
DATALINES;
1 0.5 13.6 1 2  0.5
1 8.81 13.15 15 2 0.5
1 45.2 8.95 29 2 0.5
1 34.2 5.6 36 2  0.5
1 25 6.4 43 2  0.5
1 20.6 6.5 57 2  0.5
1 0.5 8.35 71 2 0.5
1 0.5 9.55 85 2 0.5;
RUN;

PROC SORT DATA=W; BY DAY;RUN;
title h=1 f="times new roman"  "Figure 2: Plot With Annotation";

DATA anno ;
            %annomac;
            %dclanno;
            when='A';
            XSYS='2';
            YSYS='2';
            HSYS='C';
   set w;
          IF xLL=. THEN xLL=x0;
          IF yLL=. THEN yLL=y0;
            %ARROW (xLL,yLL,x0,y0,BLUE,2,1, 60,OPEN);
            %LABEL (X0,Y0,LABEL,BLUE, 0, 0, 1,ARIAL,6);
 RUN;

SYMBOL C=BLUE V=CIRCLE ;
proc gplot data=w  ANNO=anno ;
      plot y0*x0   ;
run;
QUIT;
```

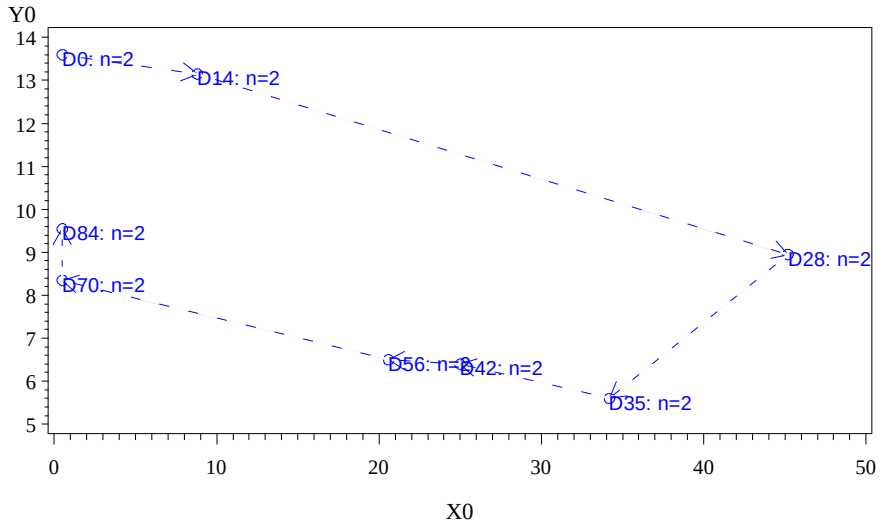
Figure 2: Plot With Annotation



There is no option in %arrow to get different line types. So to get different arrows, we have to do a little trick: cover the arrows with white dashed lines;

```
data anno ;
              %annomac;
              %dclanno;
              when='A';
              XSYS='2';
              YSYS='2';
              HSYS='C';
 set w;
        IF xLL=. THEN xLL=x0;
        IF yLL=. THEN yLL=y0;
               %ARROW (xLL,yLL,x0,y0,BLUE,3,1, 60,OPEN);
               %LINE (xlL,ylL,x0,y0, white,3, 2);
               %LABEL (X0,Y0,LABEL,BLUE, 0, 0, 1,ARIAL,6);
 RUN;

title h=1 f="times new roman" "Figure 3: Plot With Annotation-Change Line Types";

SYMBOL C=BLUE V=CIRCLE ;
proc   gplot data=w  ANNO=anno ;
       plot y0*x0   ;
run;
QUIT;
```

Figure 3: Plot With Annotation-Change Line Types

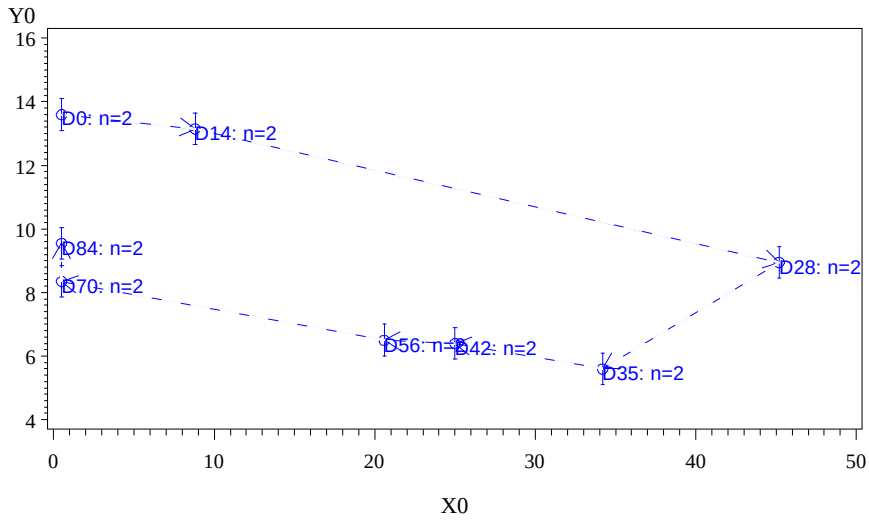


In order to add error bars, we have to calculate standard errors first, and then draw the error bars with SAS annotation. Two more variables have to be calculated to add the error bars.

```
* To make the data for the mean and error bar;
data w  ;
   set w;
               upper=y0+se;
               lower=y0-se;
                   xstart=x0-0.1;
                   xend  =x0+0.1;
run;

data anno ;
             %annomac;
             %dclanno;
             when='A';
             XSYS='2';
             YSYS='2';
             HSYS='C';
   set w;
             IF xLL=. THEN xLL=x0;
             IF yLL=. THEN yLL=y0;

               %ARROW (xLL,yLL,x0,y0,BLUE,3,1, 60,OPEN);
               %LINE (xlL,ylL,x0,y0, white,3, 2);
               %LABEL (X0,Y0,LABEL,BLUE, 0, 0, 1,ARIAL,6);
               %LABEL (X0,Y0,'H',BLUE, 0, 0, 1,special,5);
               %LINE (xstart,lower,xend,lower,blue,1, 1);
               %LINE (xstart,upper,xend,upper,blue,1, 1);
               %LINE (x0,lower,x0,upper,blue,0.1, 1);

 RUN;
```

```
title h=1 f="times new roman"
"Figure 3-2: Plot With Annotation-Change Line Types";

SYMBOL C=BLUE V=circle i= ;
axis1 order=(4 to 16 by 2);

proc gplot data=w    anno=anno ;
      plot y0*x0   /vaxis=axis1;;
run;
QUIT;
```

Figure 3-2: Plot With Annotation-Change Line Types



As we can see from the series of codes above:

1) We have to take care of many parameters in the code using SAS annotations. When there are several treatment groups, we have to add in even more flexible codes which entail a lot of efforts and testing activities.

2) %arrow, up to now, does not have the parameter for arrows with different line types. If we want to use dashed lines, we will have to code extra steps to cover up the arrow line with %line.

So we determined to find the most flexible and easiest code for the special plots. Can SAS new procedures help us?

A CLOSE LOOK AT SGPLOT (VECTOR STATEMENT)

SAS 9.2 proc sgplot have a lot of statements for us to play with. One of them is Vector statement.

The syntax: **VECTOR X= numeric-variable Y= numeric-variable </ option(s)>;**

- **Vector options:**

  ARROWDIRECTION= OUT | IN | BOTH

  ARROWHEADSHAPE= shape

  DATALABEL <= variable>

  LINEATTRS= style-element <(options)> | (options)

  NOARROWHEADS

  NOMISSINGGROUP

  XORIGIN= numeric-value | numeric-variable

  YORIGIN= numeric-value | numeric-variable

- **Plot options:**

  GROUP= variable

  LEGENDLABEL= "text-string"

  NAME= "text-string"

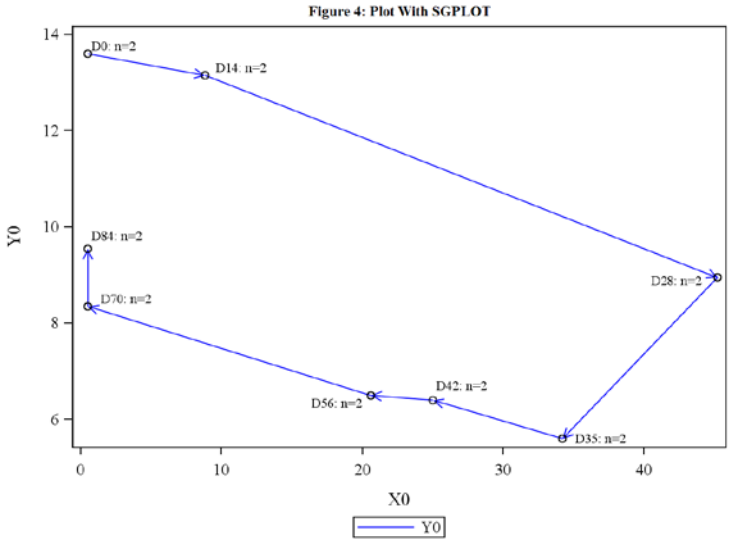
  TRANSPARENCY= numeric-value

  X2AXIS

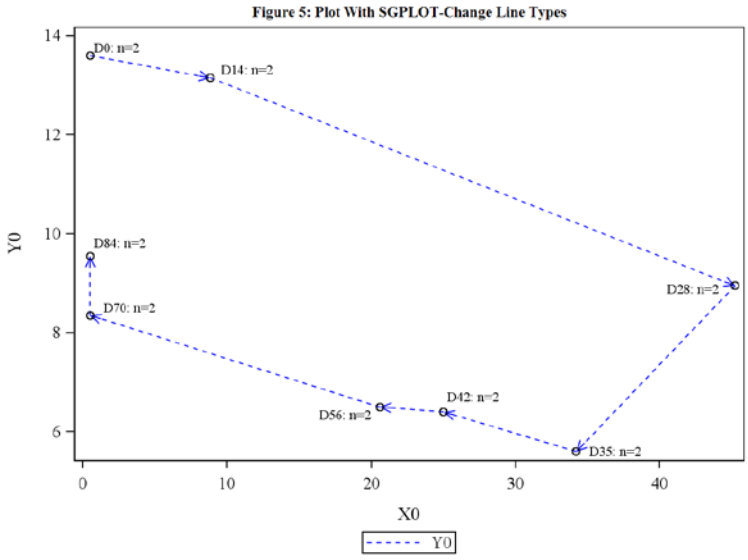
  Y2AXIS

The statement works very much like SAS annotation, it enables programmer to put arrows or lines wherever they want on plots. The same as SAS annotations, locating the starting point and end point to be drawn is critical.

```
*The simple code using  sgplot;

title h=1 f="times new roman"
"Figure 4: Plot With SGPLOT";

proc sgplot data=w;
VECTOR X=x0  Y=y0 /
XORIGIN=xlL YORIGIN=ylL
name="1" DataLabel=LABEL
lineattrs=(color=blue);
SCATTER X=x0 Y=y0;
Keylegend "1";
RUN;
```

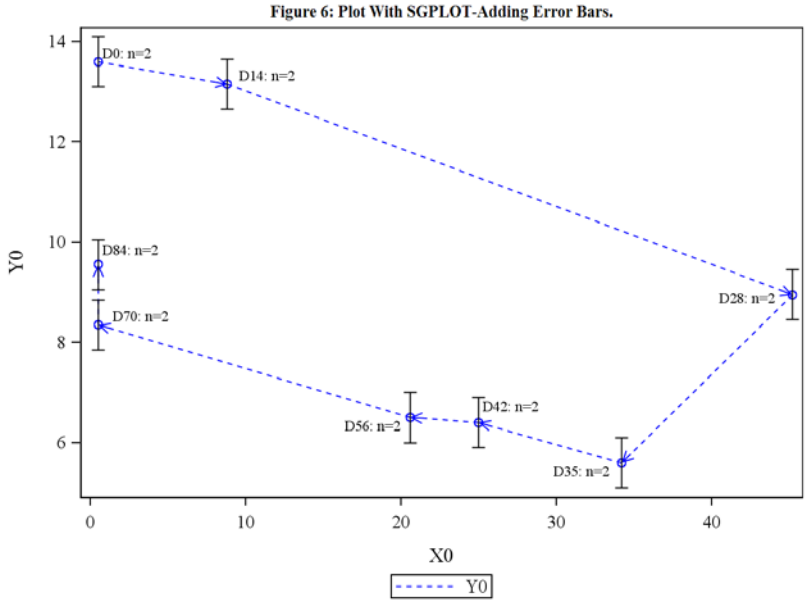



```
*It only took one option to have
arrow plots with dashed lines.
title h=1 f="times new roman"
"Figure 5: Plot With SGPLOT-
Change Line Types";

proc sgplot data=w;
VECTOR X=x0  Y=y0 /
XORIGIN=xlL YORIGIN=ylL
name="1" DataLabel=LABEL
lineattrs=(color=blue)
lineattrs=
( color=blue pattern=2);
;
SCATTER X=x0 Y=y0;
Keylegend "1";
RUN;
```



We can even easily make error bars with sgplot/scatter options:

```
*The code to generate plot with
bars;

title h=1 f="times new roman"
"Figure 6: Plot With SGPLOT-
Adding Error Bars.";

proc sgplot data=w
TMPLOUT="H:\TMPL.LST";
      VECTOR X=x0  Y=y0 /
XORIGIN=xlL YORIGIN=ylL
name="1" DataLabel=LABEL
lineattrs=(color=blue
pattern=2);
      SCATTER X=x0
Y=y0/markerattrs=(color=blue)
      yerrorupper=upper
yerrorlower=lower;
      Keylegend "1";
RUN;
```



Beautiful!! The most important thing is that the code is so straight forward that code management and validation are EASY. The wide range of options Vector enables us to generate plots with different colors, lines, arrow types and arrow bars. But when we need to change the data label attributes or error bar attributes, we will have to know a little about GTL (**Graph Template Language**).

Here is the easy way of using GTL:

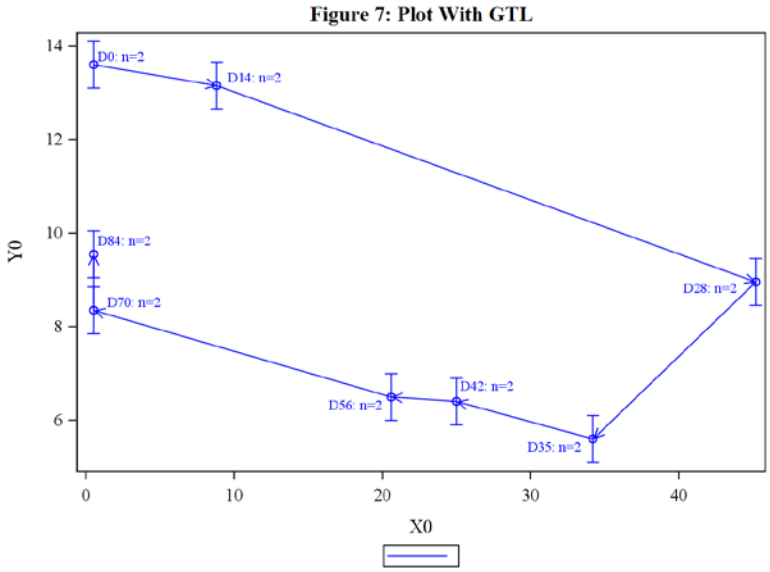
```
proc sgplot data=W TMPLOUT="C:\TMPL.LST";
      VECTOR X=X0  Y=Y0 /  XORIGIN=XLL YORIGIN=YLL name="1" DataLabel=LABEL
      lineattrs=(color=blue);
      SCATTER X=X0 Y=Y0;
      Keylegend "1";
RUN;
```

Here is SAS generated code:

```
proc template;
define statgraph sgplot;
begingraph;
layout overlay;
   VectorPlot X=X0 Y=Y0 XOrigin=XLL YOrigin=YLL / Lineattrs=( Color=BLUE)
DataLabel=label LegendLabel="Y0" NAME="1";
   ScatterPlot X=X0 Y=Y0 / primary=true LegendLabel="Y0" NAME="SCATTER";
   DiscreteLegend "1" / Location=Outside;
endlayout;
endgraph;
end;
run;
```

Now we add in the option (`datalabelattrs`) in vectorplot statement:

```
proc template;
      define statgraph sgplot;
      begingraph;
      ENTRYTITLE
 "Figure 7: Plot With GTL";
      layout overlay;
         VectorPlot X=x0 Y=y0
XOrigin=xlL YOrigin=ylL /
Lineattrs=( Color=BLUE)
DataLabel=label LegendLabel=" "
NAME="1"
datalabelattrs=(color=blue);;
         ScatterPlot X=x0 Y=y0 /
primary=true
Markerattrs=( Color=BLUE)
LegendLabel=" " NAME="SCATTER"
YERRORUPPER=upper
YERRORlower=lower
errorbarattrs=(color=blue);;
         DiscreteLegend "1" /
Location=Outside;
      endlayout;
      endgraph;
      end;
run;
```



CONCLUSION

We have compared the methods of generating hysteresis plots. We believe SGPLOT and GTL will give us the most flexibility and controllability in generating hysteresis plots using SAS.

REFERENCES

1. SAS/BASE Software: Version 9.2, SAS® Institute Inc., Cary NC
2. SAS/GRAPH 9.2: Statistical Graphics Procedures Guide

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

<Deli Wang>
<Deli.wang@hotmail.com>
<Trisha Lin>
<Lin.trisha@yahoo.com>

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies.