Paper 213-2009
Wandering Cross Reference Lines in PROC GPLOT

Sharon Avrunin-Becker, Westat, Rockville, MD
Marie Byrd Alexander, Westat, Rockville, MD

ABSTRACT

PROC GPLOT is an interesting and somewhat mysterious SAS® procedure. New users of PROC GPLOT often find the procedure useful in producing simple graphs, but find it intimidating when trying to enhance and guide the visual interpretation of the graph. One type of plot involves the use of vertical and/or horizontal reference lines to emphasize the plotted points on the graph. Usually, when the horizontal (HREF) and vertical (VREF) reference lines are defined in the PLOT statement, a specific number is referenced to create the lines (e.g. HREF=10).

What do you do if you have a graph where you want to have horizontal, vertical, or both, reference lines placed dependant on the data itself? This paper will demonstrate how to turn your reference lines parameters into macro variables and how to call the same PROC GPLOT routine to create unique and individualized graphs for each subject. When we show you the results for 8 sample graphs, you will be able to see the reference lines actually "moving" based on the data.

INTRODUCTION

You have just been called by a puppy weight loss center and told that they want to see a graph showing the height and starting weight of each of their clients. They want to be able to visually show how much weight each puppy has to lose. They have requested that you set reference lines based on the ideal weight for the puppy which is dependent on the puppy's current height. This will enable the owners to quickly see how much their puppy needs to lose to reach its goal.

SAMPLE DATA

```
data dogs;
length  breed name $20. height goalwght currweight 3. ;
input breed name height goalwght currweight ;
cards;
      Beardie Brownie 21 50 75
      Shi-Tzu Cupcake 10 7 11
      Havanese Fudge 11 10 14
      Bullmastiff Cookie 27 120 130
      Greyhound Hugs 27 60 70
      Greyhound Payday 30 70 85
      Greyhound Snickers 29 65 80
      Greyhound Kisses 28 63 75
run;

proc sort data=dogs out=dog2; by breed name; run;
```

REFERENCE LINES

To start, you might be wondering how to create reference lines in PROC GPLOT. The usual method is just stating in the GPLOT procedure what you wish the HREF and VREF to be.

In this example, we want to set the HREF and VREF for the standards of all Greyhounds since there are several Greyhound clients. We know the standard weight should be set to 65 pounds and the standard height should be 28 inches.

```
proc sort data=dogs out=gdog; by name; where breed='Greyhound'; run;

symbol1 v=dot c=blue pointlabel=(color=blue '#currweight');

axis1 label=(h=1 'Dog Weight (pounds)') order=(0 to 130 by 5) ;
axis2 label=(angle=90 h=1 color=black 'Dog Height (inches)') order=(0 to 30 by 2);

title2 'Graph Showing Greyhound Standard Height and Weight';
```
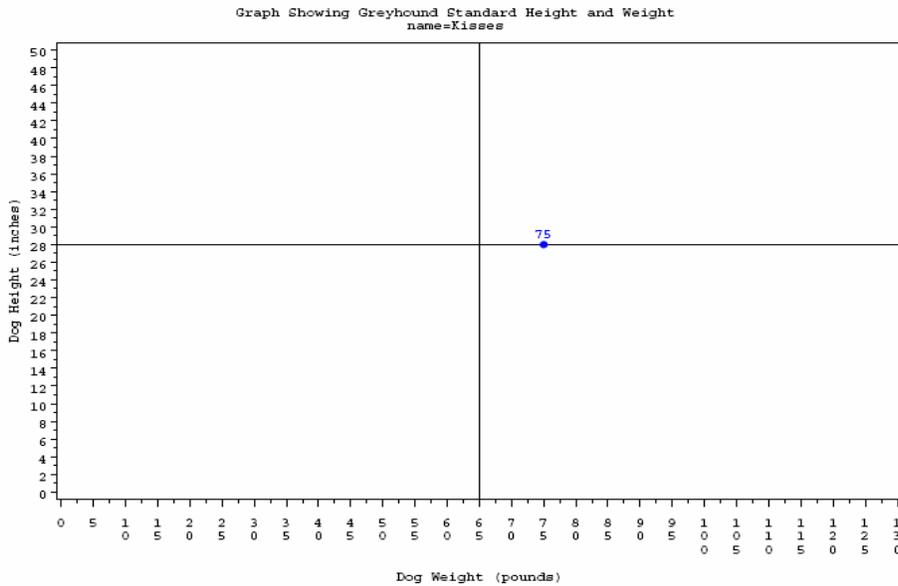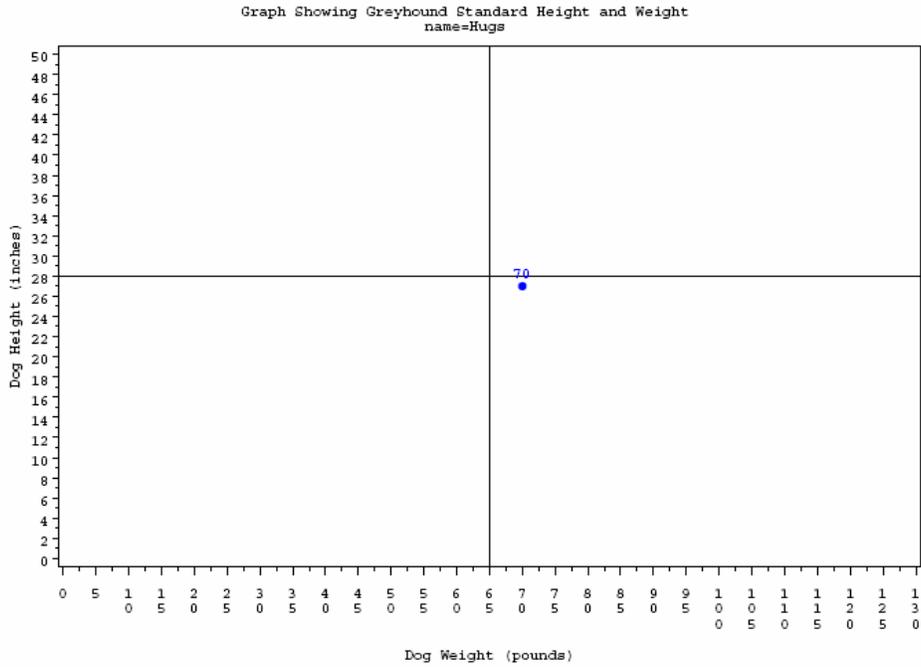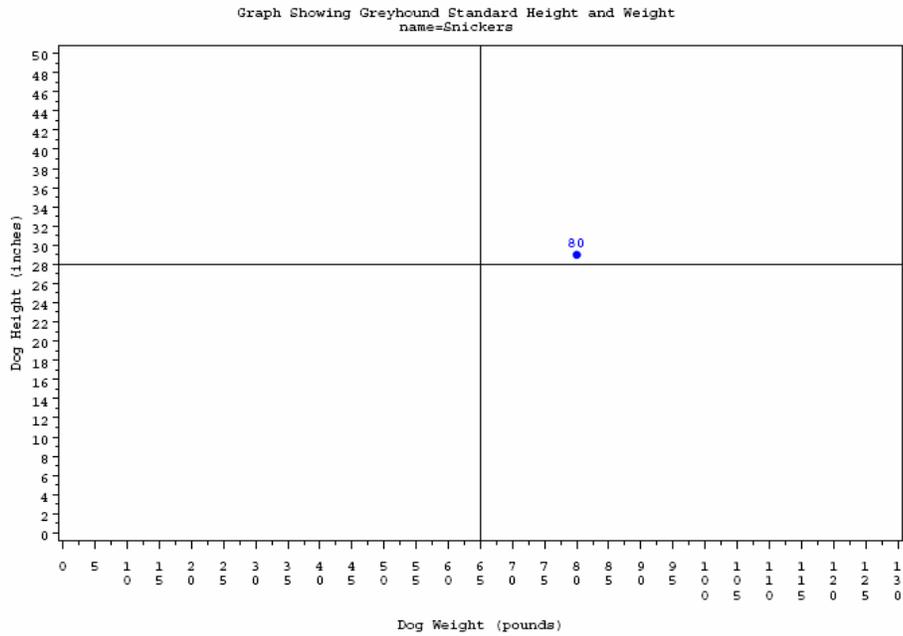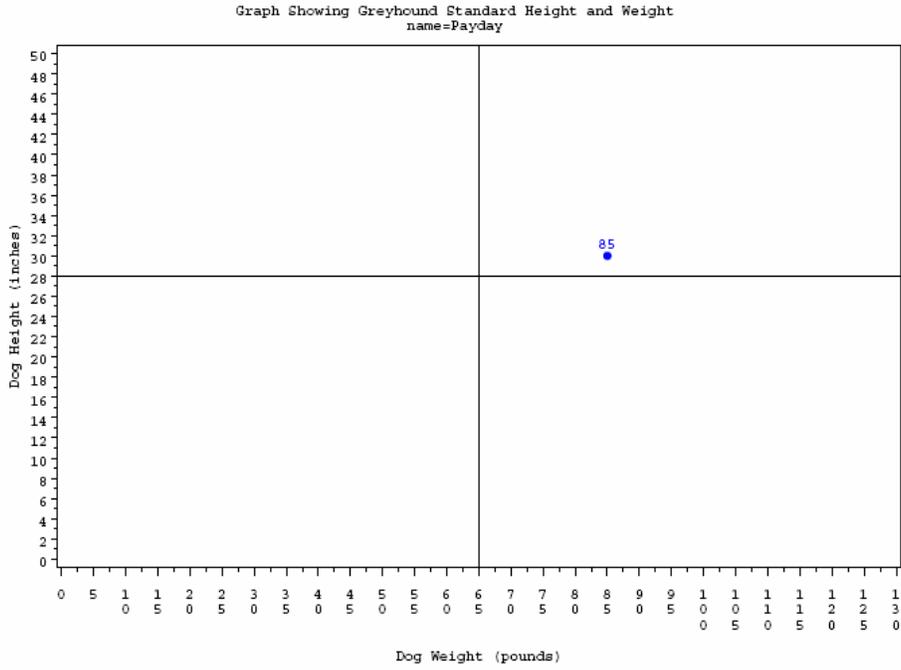
```
proc gplot data=gdog;
by name;
plot height*currweight / haxis=axis1 vaxis=axis2 href=65 vref=28;
run;
quit;
```

Here are the results of the GPLOT:

Graph Showing Greyhound Standard Height and Weight
name=Payday

Graph Showing Greyhound Standard Height and Weight
name=Snickers

3

WANDERING REFERENCE LINES

The first step to creating reference lines that wander is to use a CALL SYMPUT statement to turn the desired values into macro variables. In this case we are using the value of the variable GOALWGHT as our horizontal reference line and the value of HEIGHT will become our vertical reference line.

```
data dog3;
  set dog2 ;
  if _n_=&i;
  call symput('hortref',goalwght);
  call symput('vertref',height);
run;
```

CREATE MACRO

However, just using CALL SYMPUT will not give you the desired results you are looking for because the program will not know that it needs to be looking at other observations. To be able to have your HREF and VREF accurately wander for each observation, you will need to create a macro with a DO loop to get it the data to repeat and reset for each puppy so we can see the height and weight standards for each breed.

To trigger the DO loop we need to determine the number of observations.

```
data _null_;
  set dog2 end=eof;
  by breed name;
  if first.name then count+1;
  if eof then
  call symput('total',trim(left(count)));
 run;
```

After the number of observations have been determined we can setup the macro, starting with the DO loop to cause the GPLOT procedure to execute for each puppy listed in the dataset.

```
%macro gplot;
%do i=1 %to &total;
```

Here is the CALL SYMPUT as explained above. The variables BREED and NAME have been added for use in the TITLE statements.

```
data dog3;
  set dog2 ;
  if _n_=&i;
  call symput('hortref',goalwght);
  call symput('vertref',height);
  call symput('whichbreed',breed);
  call symput('whichname',name);
run;

symbol1 v=dot c=blue pointlabel=(color=blue '#currweight');;

axis1 label=(h=.5 'Dog Weight (pounds)') order=(0 to 130 by 5) ;
axis2 label=(angle=90 h=.5 color=black 'Dog Height (inches)') order=(0 to 30 by 2);

title2 'Graph Showing Puppy Starting Weight And Goal Weight';
title3 "Dog Breed= &whichbreed";
title4 "Dog Name= &whichname";
```
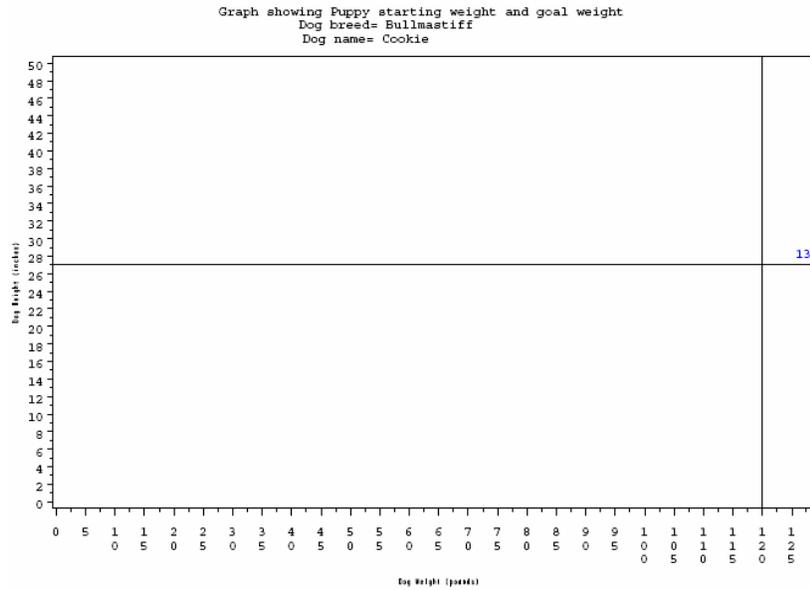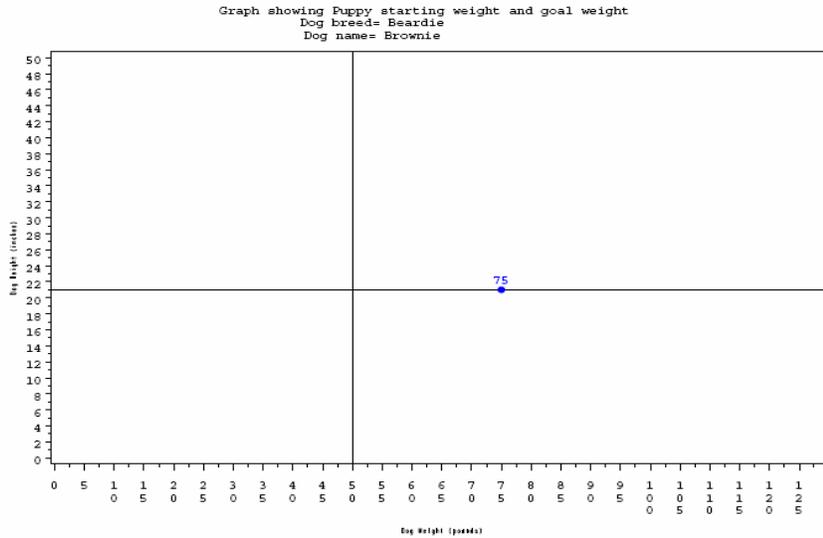
Here the GPLOT procedure is called and notice the &HORTREF and &VERTREF in the statement where previously the HREF and VREF values had been hardcoded directly into the code.
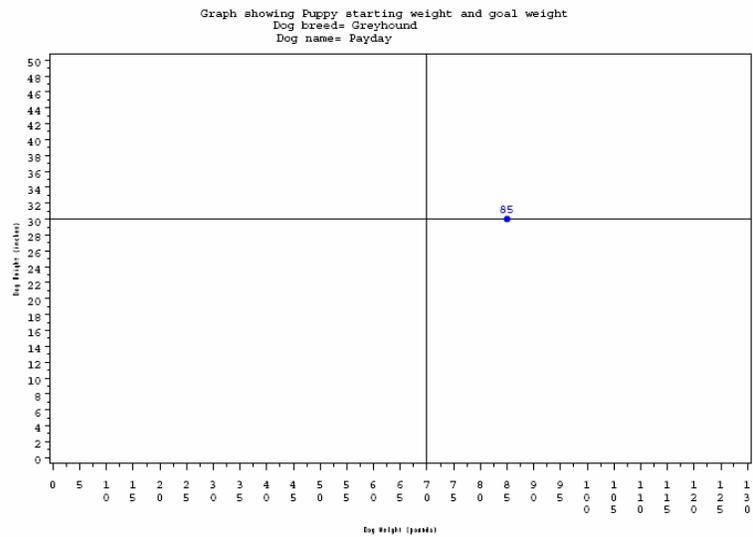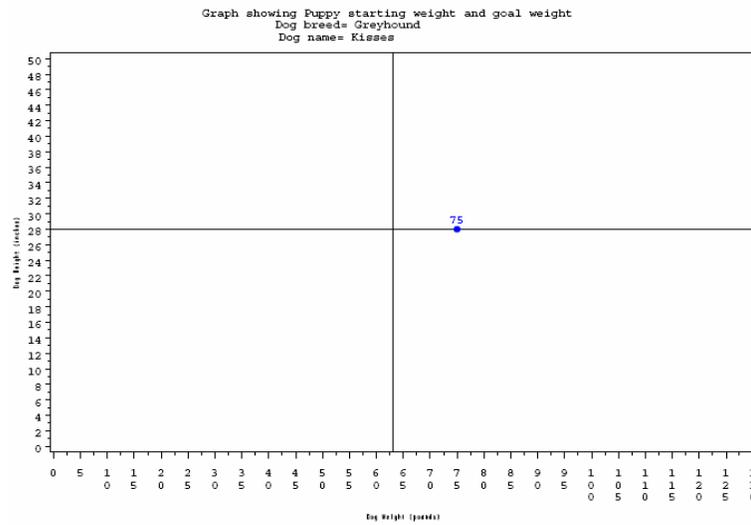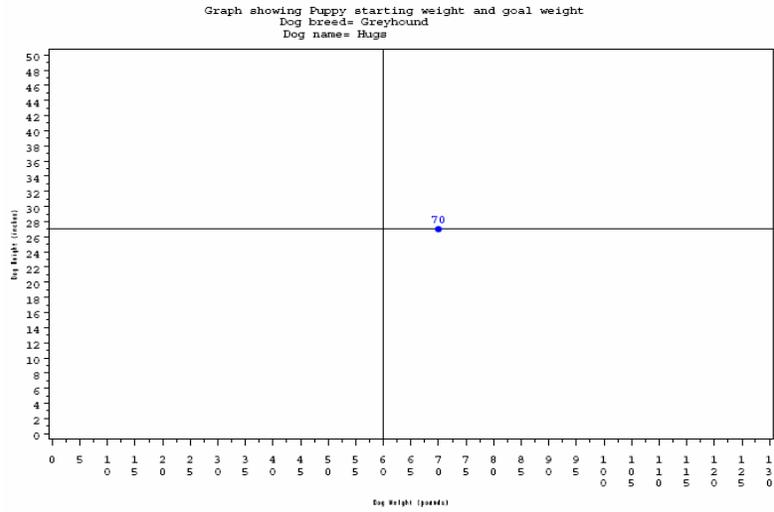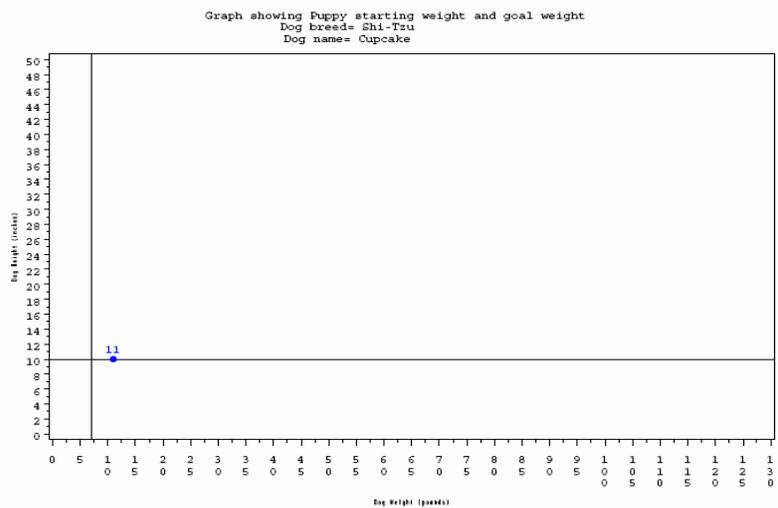
```
proc gplot data=dog3;
by breed name;
 plot height*currweight / haxis=axis1 vaxis=axis2 href=&hortref vref=&vertref;
 run;
quit;
%end;
%mend gplot;
%gplot
```

Here are the results of the GPLOT:

Graph showing Puppy starting weight and goal weight
Dog breed= Greyhound
Dog name= Hugs

Graph showing Puppy starting weight and goal weight
Dog breed= Greyhound
Dog name= Kisses

Graph showing Puppy starting weight and goal weight
Dog breed= Greyhound
Dog name= Payday

Graph showing Puppy starting weight and goal weight
Dog breed= Greyhound
Dog name= Snickers

Graph showing Puppy starting weight and goal weight
Dog breed= Havanese
Dog name= Fudge

Graph showing Puppy starting weight and goal weight
Dog breed= Shi-Tzu
Dog name= Cupcake

7

CONCLUSION

As you begin to explore the world of PROC GPLOT, you no longer need to be limited in how you decide to use both horizontal and vertical reference lines. By using a CALL SYMPUT within a macro DO loop you have endless possibilities on which data points you wish to emphasize.

ACKNOWLEDGMENTS

Many thanks to Rick Mitchell, Mark Friedman, Mike Rhoads, Ed Heaton, Michael Raithel, and Duke Owen.

DISCLAIMER

The contents of this paper are the work of the author(s) and do not necessarily represent the opinions, recommendations, or practices of Westat.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. For more information contact:

Sharon Avrunin-Becker
Westat
1650 Research Blvd. – WB432
Rockville, MD 20850
Work Phone: (301) 738-8384
Email: SharonAvruninBecker@westat.com

Marie Byrd Alexander
Westat
1650 Research Blvd. – WB472
Rockville, MD 20850
Work Phone: (301) 517-4137
Email: MarieAlexander@westat.com

TRADEMARKS

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.