Paper 130-2007

The ABC'S of graphical data analysis using SAS/GRAPH®

Hany Aboutaleb, Biogen Idec, Cambridge, MA

Abstract:

No single statistical tool is as powerful as a graph. Using graphs we can display a large amount of information, look for patterns and relationships, confirm or disprove the expected hypothesis and discover new ideas. This presentaion will describe and illustrate a graphical methodology for data analysis using basic rules and principles. The following dossier of statistical graphs illustrate the methods and principle elements of graphing data using SAS/GRAPH® software. The SAS system and SAS/GRAPH® software provide basic tools for statistical analysis and graphical display.

Introduction:

This presentaion is divided into several sections and is designed to outline the principles and methodology for graphical data analysis, describe the SAS/GRAPH® environment, and illustrate the various graphical elements using the SAS system and SAS/GRAPH® software.

- Clear Data Presentation
- Clear Understanding
- Scale /Ranges
- General Strategy

The SAS/GRAPH® software is used with data from SAS data sets or existing data with a PROC step to generate graphics output which can be displayed, written to a file, printed or plotted. Various methods will be demonstrated in SAS/GRAPH® to produce graphs and illustrate the basic rules for graphical data analysis.

BASIC ELEMENTS / RULES FOR GRAPHICAL DATA ANALYSIS

1. **Clear Data Presentation:**

   - Make the data stand out by using a graphical element that is visually projected (i.e., line plot without symbol). Avoid extra elements of the graph to interfere with the data.
   - Connecting plotting symbols by lines is useful to track the movement of the values. Avoid obscuring the symbol with the lines or overlapping symbols so all data can be seen.
   - Use a pair of scale lines for the horizontal and vertical scales and position the data region interior of the rectangle. Use a reference line when there is an important value that must be seen across the entire graph. Place tick markers outside of the data region. Avoid clutter in the data region and minimize the number of tick marks.
   - Do not graph too much or show everything that comes to mined on a single graph. Different graphical elements in the data region obscure one another.
   - Do not allow data labels, notes, keys, and markers in the data region to interfere with the data or to clutter the graph. Put keys and marker outside the data region and put notes in the legend or footnote text.
   - Plotting symbols must be visually distinguishable and clarity preserved under reduction and reproduction so all the data can been seen.

2. **Clear Understanding:**
   - Put major conclusions into graphical form. Make legends comprehensive, informative, independent so that they describe everything that is graphed, and draw attention to the important features of the data.
   - Error bars should be clearly explained and communicate meaningful information relating to the data. Avoid ambiguous descriptions.
   - Use logarithms of a variable when percent change is important. The scale label should correspond to the tick mark labels.
   - Graphs should be carefully checked for errors.
   - Graphs designed for clarity describe everything that is graphed, enable important data to stand out and communicate conclusions drawn from the data.

3. **Scales / Ranges:**
   - Choose the range of the tick marks to include or nearly include the data.
   - Choose the scales so that the data fill up as much of the data region as possible.
   - Do not insist that zero always be include on a scale showing magnitude, because sometimes it ruins the resolution of the data on the graph.
   - Choose appropriate scale when graphs are compared.
   - Use a logarithmic scale when it is important to understand percent change or multiplicative factors.
   - Showing data on a logarithmic scale can improve resolution.
   - Use a scale break only when necessary. If a break cannot be avoided, use a full scale break. Do not connect numerical values on two sides of a break.

4. **General Strategy:**
   - A large amount of quantitative information can be packed into a small region.
   - Graphing data should be iterative, experimental process.
   - Graph data two or more times when needed.
   - Many useful graphs require careful, detailed study.
   - Include number of observation when graph is presenting summary data.

Producing Graphics Output in The SAS System:

The SAS system and SAS/GRAPH® software provide the basic tools for statistical analysis and graphical display. When we use SAS/GRAPH® software, we run programs to produce graphics. Once they are created we can:

- Display graphics on the monitor.
- Print graphics on a printer or plotter.
- Store graphics, either in a host file or SAS catalog.

To display the graphs on the monitor, you must first select a SAS/GRAPH® device driver .

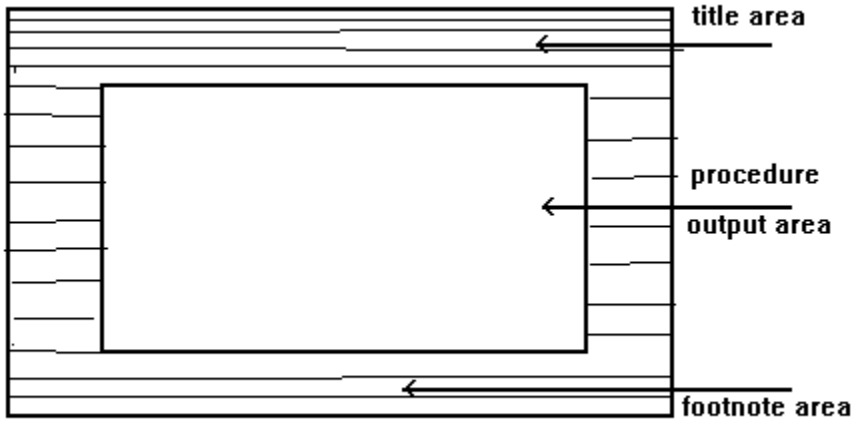
1. Begin a new interactive SAS session using the SAS system or SAS Enterprise Guide.
2. You can write a list of device drivers to a file by entering the following program:

```
proc printto print='gdevice.txt';
     proc gdevice nofs;
      list _all_;
```

3. A list of thedevice drivers (i.e., up to 600 lines)will be saved to the text file.
4. Scan the list and choose the device drivers that match your monitor and printer (i.e., you can check Biogenidec setup macro program).
   (example (SAS Windows): display - win, printer – winplot target - cgmof97l).
   (example (SAS UNIX): display - ActiveX, printer – sasemf target - cgmof97l).

Specifying a target device ensures the way the graphs display on your device is as close as possible to the appearance of the graph.

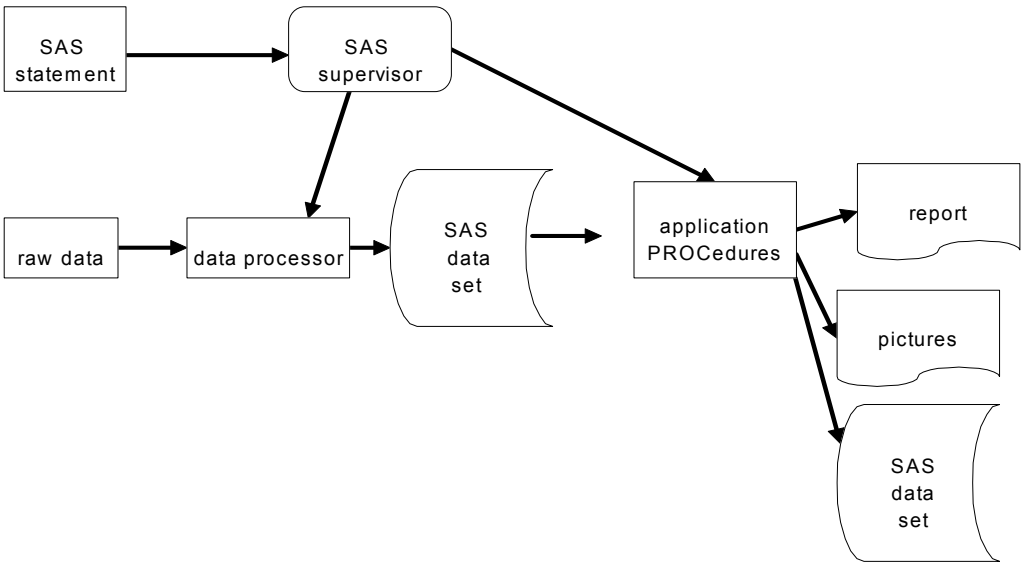
How Graphics Are Drawn:



1. Titles are positioned at the top of the graphics output area.
2. Footnotes are positioned at the bottom of the graphics area.
3. By default, legends appear below the axis area of a graph and above the footnote area.
4. the graph uses the area left after space for the titles, footnotes, and legend is reserved. This area is the procedure output area.

   because space for titles, footnotes and possibly the legend is reserved before a graph is drawn, you should carefully choose the heights and units for text.

HOW SAS/GRAPH fits into the SAS System:



SAS/GRAPH® produces graphs by using data from SAS data sets or with graphics generated in a PROC run on existing data which can then be displayed in the graph window, written to a file or printed.

A SAS/GRAPH® program is made up of these parts:

1. FILENAME statements: for external files uses for input and output.
2. LIBNAME statements: for SAS libraries to use for input or output.
3. GOPTIONS statements: to set up global characteristics of a graph and can control where the output is sent.
4. global statements: to set up the titles, footnotes, legend, axes, symbols, and patterns.
5. procedure statements: determine what data sets are used , what catalog to use for output, and what kind of graph is produced.

Goptions statement:

RESET=GLOBAL
cancels all currently defined AXIS, FOOTNOTE, LEGEND, PATTERN, SYMBOL, and TITLE definitions. It is useful to rest Goptions every time you write program to display graph.
GUNIT=PCT
set options in SAS/GRAPH procedures and global statements use units of percent of the graphics output area unless other units are used.
ROTATE=LANDSCAPE
set graphs to use landscape orientation.
CBACK=WHITE
set the background color of the monitor is white.
HTITLE=4
set the text height for the first title is 4 (in units of percent of the display height).
HTEXT=3
set the text height for all text on graphs is 3 (units of percent of the display height).
FTEXT=DUPLEX
set the text font is duplex for all text on graphs.
HPOS=80,VPOS=40
set the graphs to use the appropriate number of rows and columns.
With the global statements you can apply the basic rules and principles for producing graphs as follows:

Titles and Footnotes:

Can appear anywhere in the program and once they are defined , the titles and footnotes remain in effect until they are canceled the SAS session is ended. On graphs, you can use up to ten titles and ten footnotes . To change a title or footnote that is already defined , issue another TITLE statement that has the same number as the title you want to change. You can control the font and the height of the characters by FTEXT= and HTEXT options that define in the GOPTIONS statement but by using certain options within TITTLE and FOOTNOTE statements , you can override the effect of those options for more

AXIS:

Axis statements give the user ability to control the location, values, and appearance of the axes of the charts. You can have up to 99 axis, you can control scale, order of the data, location and appearance of the axes lines and tick marks, and the text appearance of the axes labels and major tick marks. Axes are automatically scaled and labeled.

LEGEND:

The legend statements give the user ability to control the location and appearance of legends on a charts. You can have up to 99 legends , you can control the position and appearance of the legend, and the text and appearance of the legend label, you cancontrol the appearance of legend entries like the size and shape of the legend values and their text labels.

PATTERN:
The pattern statements define the characteristics of a patterns used in a graphs. You can have up to 99 patterns, you can control the type of the pattern, and color, and repeat the pattern. There are three dfferent types of patterns, bar and block patterns,map and plot patterns, and pie and star pattern. The procedure uses the pattern you create or can generates them as needed and assigns them to the graph by default.

SYMBOL:
The symbol statements define the characteristics of the symbols that display the data plotted by GPLOTprocedure. You can have up to 99 symbol. You can control the appearance of plot symbols and plot lines including bars, boxes, confidence limit lines and area fill, interpolation methods, and how plots handle data out of range.

Annotate Facility:
The annotate facility customizes procedure output and enhances graphics by adding text or some elements to the graphics output of procedures. This facility can also be used to construct custom graphics output by itself.

Macro Language:
Use SAS macro language so that the programs are flexible enough to accept changes in the device or the data. The efficiency of your program could be improved by adding macros.

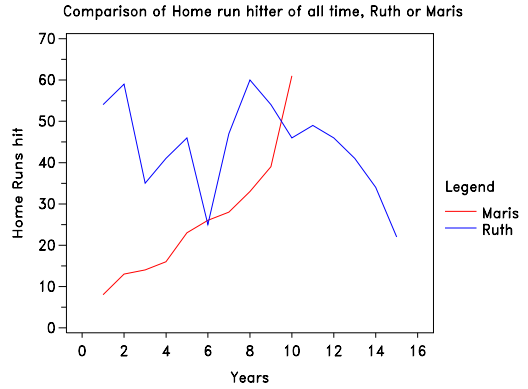
Procedure Statement:
To draw graphs, within a procedure certain statements and options are used.

Conclusion
This paper reveal that like good writing, producing an effective graphical display requires an understanding of what aspects of the data emphasized to designing graphical displays that communicate effectively.

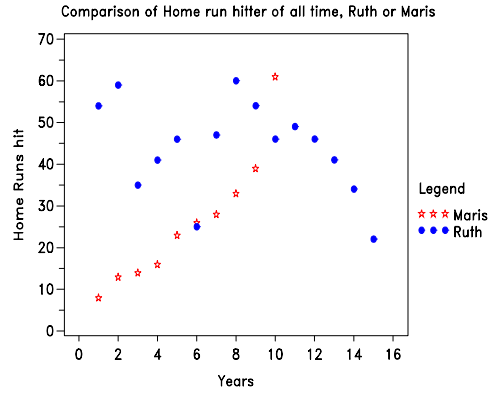
References
[1]Chambers, John M. and Cleveland, William S. and Kleiner,Beat and Tukey,Paul A. (1983) *Graphical Methods for Data Analysis,* Wadsworth & Brooks/Cole statistics/probability series.
[2]Cleveland, William S. (1985), *The Elements of Graphing Data*, Monterey:Wadsworth Advanced Books and Software.
[3]SAS/GRAPH *Software: Introduction, Version 6, First Edition*, SAS Institute Inc. , Cary, NC
[4]SAS/GRAPH *Software: Reference, Version 6, First Edition, Volume 1 and volume 2*, SAS Institute Inc. , Cary, NC

CONTACT:
Your comments and questions are valued and encouraged.
Contact the author at:
Hany Aboutaleb
Biogen Idec.
14 Cambridge Center
Cambridge MA 02142
Tele. #: (617) 914-7125
Fax: (617) 679-3280 Internet: hany.aboutaleb@biogenidec.com

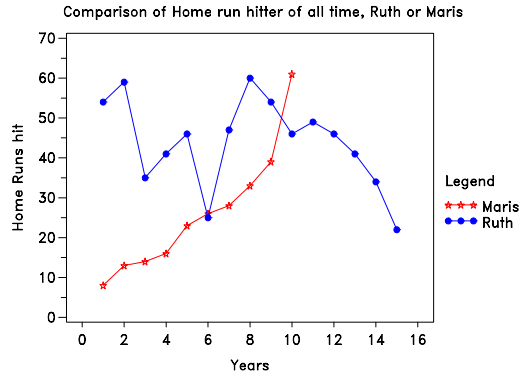
Graphical methods



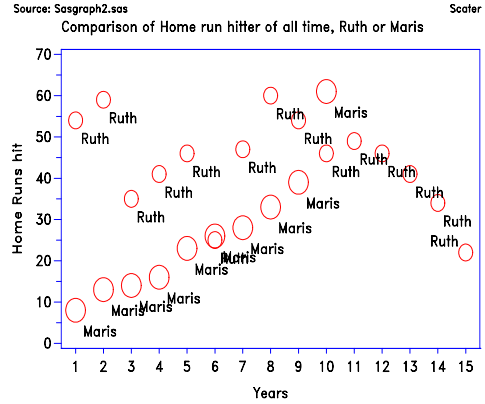
Source: Sasgraph1.sas Line plot without symbol



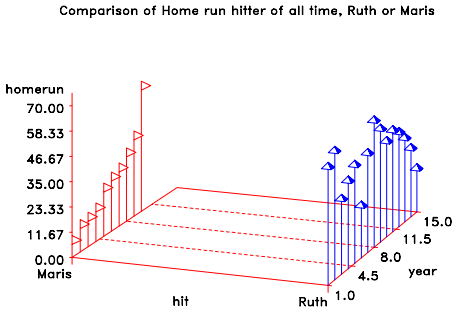
Source: Sasgraph2.sas Scater plot



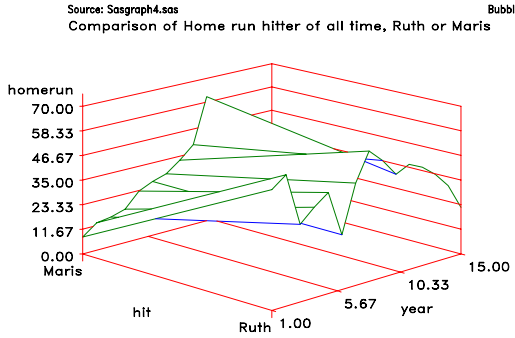
Source: Sasgraph3.sas Line plot with symbol



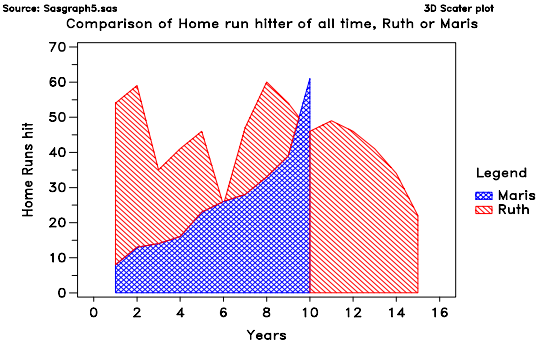
Source: Sasgraph4.sas Bubble plot



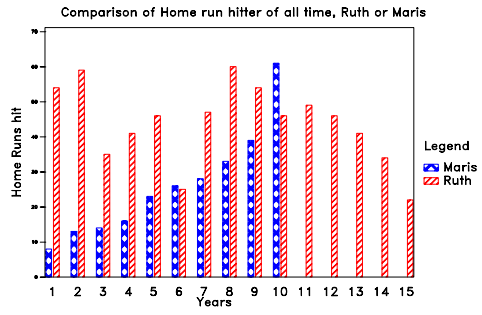
Source: Sasgraph5.sas 3D Scater plot



Source: Sasgraph6.sas 3D line plot



Source: Sasgraph7.sas line with fill area plot



Source: Sasgraph8.sas Bar plot

You can recognize which graphical methods are capable of focusing on detail (i.e. line plot)

Appendix – Data Sets: Home run hitter of all time, Ruth or Maris

Hitter	Homerun	Years
Maris	8	1
Maris	13	2
Maris	14	3
Maris	16	4
Maris	23	5
Maris	26	6
Maris	28	7
Maris	33	8
Maris	39	9
Maris	61	10
Ruth	54	1
Ruth	59	2
Ruth	35	3
Ruth	41	4
Ruth	46	5
Ruth	25	6
Ruth	47	7
Ruth	60	8
Ruth	54	9
Ruth	46	10
Ruth	49	11
Ruth	46	12
Ruth	41	13
Ruth	34	14
Ruth	22	15

```
/* Create the SAS data set homerun */

data homerun;
  input hitter $ homerun @@;
  cards;
Ruth 54 Maris 8 Ruth 59 Maris 13 Ruth 35 Maris 14 Ruth 41 Maris 16
Ruth 46 Maris 23 Ruth 25 Maris 26 Ruth 47 Maris 28 Ruth 60 Maris 33
Ruth 54 Maris 39 Ruth 46 Maris 61 Ruth 49 Ruth 46 Ruth 41 Ruth 34 Ruth 22
;
proc sort;by hitter;
run;
data homerun;set homerun;
  by hitter;
 if hitter='Ruth' then do;
      hit=1;     home1=homerun;     shapeval='pyramid';  colorval='blue';
 end;
   else do;
      hit=2 ;     home2=homerun;      shapeval='flag' ;   colorval='red';
   end;
 if first.hitter then year=0;
  year+1;
run;
```

Appendix – SAS/GRAPH® global statements:

```
  /* set selected graphics options for the examples */
goptions reset=all  device=cgmof97l  ftext=duplex ftitle=duplex simfont=duplex display
gsfname=gout rotate=landscape gsfmode=replace  gunit=pcT htitle=6  htext=4  xmax=7 inches
ymax=5 inches;
   /* define title and footnote */
title1 c=black h=4  j=c 'Comparison of Home run hitter of all time, Ruth or Maris';
footnote1 c=black h=3 j=l 'Source: Sasgraph.sas' j=r 'Graphical methods';
  /* modify horizontal axis */
axis1 minor=none label=('Years') offset=(3,3) length=70;
  /* modify vertical axis */
axis2 minor=(number=1 height=1) major=(height=1.5)
    label=(a=90 r=0 'Home Runs hit');
  /* modify the legend */
 legend1   label =(position=top j=l h=4 c=black 'Legend')
            mode  = reserve
            across=1
            shape = symbol(6,3)
           value =(h=4  j=l c=black)
           position = (outside middle right);
```

Appendix – SAS/GRAPH® Procedure Statement:

```
/* produce a a line plot without symbol  */
      /* output file */
filename gout "Homerun1.cgm" ;
  /* modify symbol characteristics */
symbol1 interpol=join value=none color=red height=3;
symbol2 interpol=join value=none color=blue height=3;
ods rtf file="Homerun1.rtf" ;
/* define title and footnote */
title1 c=black h=4  j=c 'Comparison of Home run hitter of all time, Ruth or Maris';
footnote1 c=black h=3 j=l 'Source: Sasgraph1.sas' j=r 'Line plot without symbol';
 /* produce a a line plot without symbol  */
proc gplot data=homerun;
  plot homerun*year=hitter/ frame haxis=axis1 vaxis=axis2 legend=legend1 ;
run;
quit;
ods rtf close;
%*-------------------------------------------------------------------------------------------------------;
/* produce a Scater plot with a legend  */
      /* output file */
filename gout "Homerun2.cgm" ;
  /* modify symbol characteristics */
symbol1 interpol=none value== color=red height=3;
symbol2 interpol=none value=dot color=blue height=3;
ods rtf file="Homerun2.rtf" ;
/* define title and footnote */
title1 c=black h=4  j=c 'Comparison of Home run hitter of all time, Ruth or Maris';
```

```
footnote1 c=black h=3 j=l 'Source: Sasgraph2.sas' j=r 'Scater plot';
 /* produce a Scater plot with a legend  */
proc gplot data=homerun;
   plot homerun*year=hitter/ frame haxis=axis1 vaxis=axis2 legend=legend1 ;
run;
quit;
ods rtf close;
%*-----------------------------------------------------------------------------------------------------------;
/* produce a Line plot with symbol with a legend  */
       /* output file */
filename gout "Homerun3.cgm" ;
   /* modify symbol characteristics */
symbol1 interpol=join value== color=red height=3;
symbol2 interpol=join value=dot color=blue height=3;
ods rtf file="Homerun3.rtf" ;
    /* produce a plot with a legend  */
/* define title and footnote */
title1 c=black h=4  j=c 'Comparison of Home run hitter of all time, Ruth or Maris';
footnote1 c=black h=3 j=l 'Source: Sasgraph3.sas' j=r 'Line plot with symbol';
 /* produce a Line plot with symbol with a legend  */
proc gplot data=homerun;
   plot homerun*year=hitter
       / frame haxis=axis1 vaxis=axis2 legend=legend1 ;
run;
quit;
ods rtf close;
%*-----------------------------------------------------------------------------------------------------------;
/* produce a bubble plot  */
       /* output file */
filename gout "Homerun4.cgm" ;
ods rtf file="Homerun4.rtf" ;
/* define title and footnote */
title1 c=black h=4  j=c 'Comparison of Home run hitter of all time, Ruth or Maris';
footnote1 c=black h=3 j=l 'Source: Sasgraph4.sas' j=r 'Bubble plot';
        /* produce a bubble plot  */
proc gplot data=homerun;
   bubble homerun*year=hit
       / frame haxis=axis1 vaxis=axis2
        hminor=0 BSIZE=12 BCOLOR=RED BLABEL bscale=area
        Caxis=blue
       CFRAME=WHITE;
format hit hitter.;
run;
quit;
ods rtf close;
%*-----------------------------------------------------------------------------------------------------------;
/* produce a 3D Scater plot*/
       /* output file */
filename gout "Homerun5.cgm" ;
ods rtf file="Homerun5.rtf" ;
/* define title and footnote */
title1 c=black h=4  j=c 'Comparison of Home run hitter of all time, Ruth or Maris';
```

```
footnote1 c=black h=3 j=l 'Source: Sasgraph5.sas' j=r '3D Scater plot';
    /* produce a 3D Scater plot*/
proc g3d data=homerun;
  scatter hit*year=homerun / xticknum=5 yticknum=2 zticknum=7 zmin=0 zmax=70
        shape=shapeval color=colorval;
        format hit hitter.;
run;
quit;
ods rtf close;
%*-----------------------------------------------------------------------------------------------------------;
/* produce a 3D line plot*/
       /* output file */
filename gout "Homerun6.cgm" ;
ods rtf file="Homerun6.rtf" ;
/* define title and footnote */
title1 c=black h=4  j=c 'Comparison of Home run hitter of all time, Ruth or Maris';
footnote1 c=black h=3 j=l 'Source: Sasgraph6.sas' j=r '3D line plot';
    /* produce a 3D line plot*/
proc g3d data=homerun;
  plot hit*year=homerun/rotate=45 grid ctop=green cbottom=blue
       yticknum=2 zticknum=7 zmin=0 zmax=70 ;
  format hit hitter.;
run;
quit;
ods rtf close;
%*-----------------------------------------------------------------------------------------------------------;
/* produce a line with fill area plot*/
       /* output file */
filename gout "Homerun7.cgm" ;
  /* modify symbol characteristics */
symbol1 interpol=join value=none;
symbol2 interpol=join value=none;
  /* define pattern characteristics for area fill */
pattern1 value=m3x135 color=blue;
pattern2 value=m3n135 color=red;
ods rtf file="Homerun7.rtf" ;
    /* produce a plot with a legend  */
/* define title and footnote */
title1 c=black h=4  j=c 'Comparison of Home run hitter of all time, Ruth or Maris';
footnote1 c=black h=3 j=l 'Source: Sasgraph7.sas' j=r 'line with fill area plot';
 /* produce a line with fill area plot*/
proc gplot data=homerun;
  plot home1*year home2*year /overlay
       frame haxis=axis1 vaxis=axis2 areas=2 ;*legend=legend1 ;
run;
quit;
ods rtf close;
%*-----------------------------------------------------------------------------------------------------------;
/* produce a Bar plot*/
 /* output file */
filename gout "Homerun8.cgm" ;
```

```
  /* define pattern characteristics for area fill */
pattern1 value=X5 color=blue;
pattern2 value=R3  color=red;
  /* modify  axis */
axis1 value=(h=1.5) minor=none major=(h=1 w=3)
     label=(a=90 r=0 'Home Runs hit');
axis2 value=none label=none ;;
axis3 label=none ;
footnote1 '  ' move=(40,+6) h=4 'Years';
/* modify the legend */
 LEGEND1   LABEL =(POSITION=TOP J=L H=4  C=BLACK 'Legend')
          MODE  = RESERVE
          SHAPE = BAR(3,2)
          ACROSS=1
          VALUE =(H=4  J=L C=BLACK)
       POSITION = (OUTSIDE MIDDLE RIGHT);
 ods rtf file="Homerun8.rtf" ;
    /* produce a plot with a legend  */
/* define title and footnote */
title1 c=black h=4  j=c 'Comparison of Home run hitter of all time, Ruth or Maris';
footnote2 c=black h=3 j=l 'Source: Sasgraph8.sas' j=r 'Bar plot';
 /* produce a Bar plot*/
proc gchart data=homerun;
  vbar hitter /sumvar=homerun
             group=year
             DISCRETE
             subgroup=hitter
               RAXIS = AXIS1
               MAXIS = AXIS2
               GAXIS = AXIS3
               GSPACE=2
                SPACE=5
               frame
              legend=legend1 ;
run;
quit;

ods rtf close;
```