

# CHAPTER 29

## The G3D Procedure

---

<i>Overview</i>	<b>975</b>
<i>About Surface Plots</i>	<b>976</b>
<i>About Scatter Plots</i>	<b>976</b>
<i>Concepts</i>	<b>977</b>
<i>Parts of a Three-dimensional Plot</i>	<b>977</b>
<i>About the Input Data Set</i>	<b>978</b>
<i>Data for Surface Plots</i>	<b>978</b>
<i>Data for Scatter Plots</i>	<b>978</b>
<i>Changing Data Ranges</i>	<b>979</b>
<i>About Rotating and Tilting the Plot</i>	<b>979</b>
<i>About Controlling the Axes</i>	<b>980</b>
<i>Procedure Syntax</i>	<b>980</b>
<i>PROC G3D Statement</i>	<b>980</b>
<i>PLOT Statement</i>	<b>981</b>
<i>SCATTER Statement</i>	<b>985</b>
<i>Examples</i>	<b>994</b>
<i>Example 1: Generating a Default Surface Plot</i>	<b>994</b>
<i>Example 2: Rotating a Surface Plot</i>	<b>996</b>
<i>Example 3: Tilting Surface Plot</i>	<b>997</b>
<i>Example 4: Generating a Simple Scatter Plot</i>	<b>998</b>
<i>Example 5: Using Shapes in Scatter Plots</i>	<b>999</b>
<i>Example 6: Rotating a Scatter Plot</i>	<b>1003</b>
<i>References</i>	<b>1004</b>

---

### Overview

The G3D procedure produces three-dimensional graphs that plot one vertical variable ( $z$ ) for a position on a plane that is specified by two horizontal variables ( $x$  and  $y$ ). The coordinates of each point correspond to the values of three numeric variable values in an observation of the input data set. The observation may contain values in the form  $z=f(x,y)$  or independent values such as the altitude at a given longitude and latitude.

You can use the G3D procedure to

- produce surface plots or scatter plots
- examine the shape of your data
- observe data trends in a scatter plot without having a complete grid of  $x$  and  $y$  variable values

- produce scatter plots in which size, shape, or color represents a data class or the value of a fourth variable.

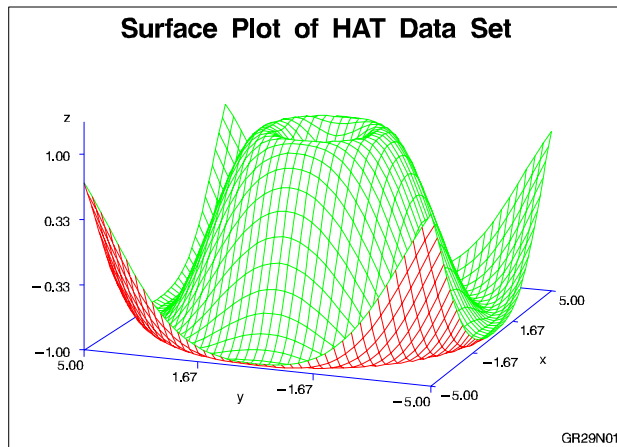
---

## About Surface Plots

Surface plots show the three-dimensional shape of your data and are useful for examining data trends. The plots represent the shape of the surface that is described by the values of two horizontal variables,  $x$  and  $y$ , and a third vertical variable,  $z$ . The values of the horizontal variables are plotted on  $x$  and  $y$  axes, which form a horizontal plane. The values of the vertical variable are plotted on a  $z$  axis, rising above that plane to form a three-dimensional surface.

Figure 29.1 on page 976 shows an example of a surface plot that uses all default settings for the plot. The axes are scaled to include the maximum and minimum values for each of the plotted variables  $x$ ,  $y$ , and  $z$ . Each variable's value range is divided into three even intervals, which form the major axes tick marks, and the axes are labeled with the names of the plotted variables or associated labels. The horizontal plane formed by the  $x$  and  $y$  axes is rotated  $70^\circ$  around the  $z$  axis and also tilted  $70^\circ$  toward you, and the plot is colored with the colors that are defined in the current colors list.

Figure 29.1 Sample G3D Surface Plot (GR29N01)



The program for this plot is shown in Example 1 on page 994. For more information on producing surface plots, see “PLOT Statement” on page 981.

---

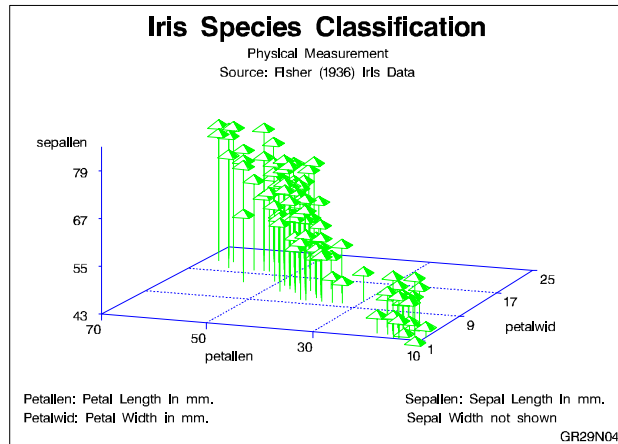
## About Scatter Plots

*Scatter plots* are three-dimensional plots that are similar to surface plots, but they represent the data as points instead of surfaces. Scatter plots show trends or concentrations in the data by classifying the data by size, color, shape, or a combination of these features. As with surface plots, the values of the  $x$  and  $y$  variables in scatter plots form a horizontal plane, and the values of the  $z$  variable rise above that plane. Rather than forming a surface, however, the values of the  $z$  variable are represented as individual symbols that are connected to the horizontal plane with lines called *needles*. Optionally, you can suppress the needles.

Figure 29.2 on page 977 shows a simple scatter plot. As with surface plots, default settings for scatter plots scale the axes to include the maximum and minimum values

for each of the plotted variables  $x$ ,  $y$ , and  $z$ , and divide each variable's value range into three even intervals to form the major axes tick marks. Default settings also rotate the horizontal plane  $70^\circ$  around the  $z$  axis and tilt it  $70^\circ$  toward you, label each axis with the name of the plotted variable or an associated label, and color the plot with colors that are defined in the current colors list. The default settings also add reference lines to the horizontal plane to mark the major  $x$  and  $y$  axes tick marks, and represent each data point with a pyramid, which is connected to the horizontal plane with a needle.

**Figure 29.2** Sample G3D Scatter Plot (GR29N04)



The program for this plot is shown in Example 4 on page 998. For more information on producing scatter plots, see “SCATTER Statement” on page 985.

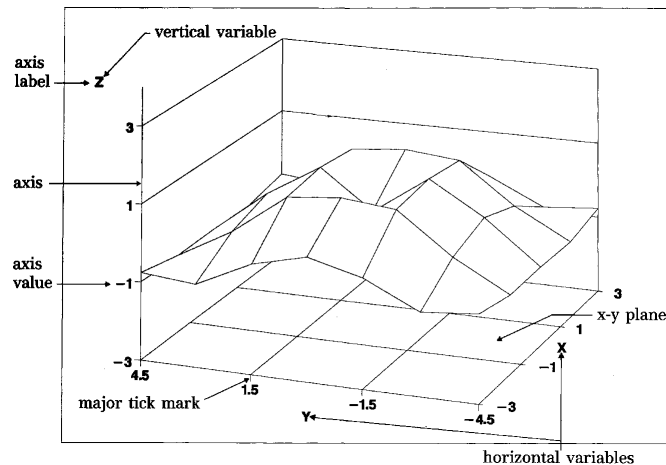
---

## Concepts

---

### Parts of a Three-dimensional Plot

Figure 29.3 G3D Procedure Terms



## About the Input Data Set

The G3D procedure requires data sets that include three numeric variables: two horizontal variables plotted on the  $x$  and  $y$  axes that define an  $x$ - $y$  plane, and a vertical variable plotted on the  $z$  axis rising from the  $\{x, y\}$  plane.

### Data for Surface Plots

For surface plots, the observations in the input data set should form an evenly spaced grid of horizontal ( $x$  and  $y$ ) values and exactly one vertical ( $z$ ) value for each of these combinations. For example, data that contains 5 distinct values for  $x$  and 10 distinct values for  $y$  should be part of a data set that contains 50 observations with values for  $x$ ,  $y$ , and  $z$ .

Only one  $z$  point is plotted for each combination of  $x$  and  $y$ . For example, you cannot draw a sphere using the PLOT statement. If there is more than one observation for a combination of  $x$  and  $y$  in the data set, only the last such point is used.

For the G3D procedure to produce a satisfactory surface plot, the data set must contain nonmissing  $z$  values for at least 50 percent of the grid cells. When the G3D procedure cannot produce a satisfactory surface plot because of missing  $z$  values, SAS/GRAPH issues a warning message and a graph may not be produced. To correct this problem, process the data set with the G3GRID procedure and use the processed data set as the input data set for G3D. The G3GRID procedure interpolates the necessary values to produce a data set with nonmissing  $z$  values for every combination of  $x$  and  $y$ . The G3GRID procedure can also smooth data for use with the G3D procedure. See Chapter 30, "The G3GRID Procedure," on page 1007 for more information on the G3GRID procedure.

### Data for Scatter Plots

An input data set for scatter plots must include at least two observations that contain different values for each of the three variables that are specified in the plot request so that the G3D procedure can scale the axes. If the data set does not meet these requirements, SAS/GRAPH software issues an error message and no graph is produced.

For scatter plots, only one  $z$  value is plotted for a combination of  $x$  and  $y$ . For example, you cannot draw a sphere using the SCATTER statement. If there is more than one observation for a combination of  $x$  and  $y$  in the data set, only the last point is

used. See “Simulating an Overlaid Scatter Plot” on page 991 for information on producing scatter plots with more than one vertical value for each  $x,y$  combination.

## Changing Data Ranges

By default for both surface plots and scatter plots, the range of the  $z$  axis is defined by the minimum and maximum  $z$  values in the input data set. Restrict or expand the range of the  $z$  axis by using the ZMIN= and ZMAX= options in the PLOT or SCATTER statement. To restrict the range of an  $x$  or  $y$  axis, use a WHERE statement in the PROC step or a WHERE or IF statement in a DATA step to create a subset of the data set.

*Note:* AXIS and LEGEND definitions are not supported by the G3D procedure. Use the Annotate facility or TITLE, FOOTNOTE, and NOTE statements to produce legends, tick mark values, and axis labels. See “About Controlling the Axes” on page 980 and “SCATTER Statement” on page 985 for information on controlling axis labels and tick mark values with PLOT statement and SCATTER statement options.  $\Delta$

---

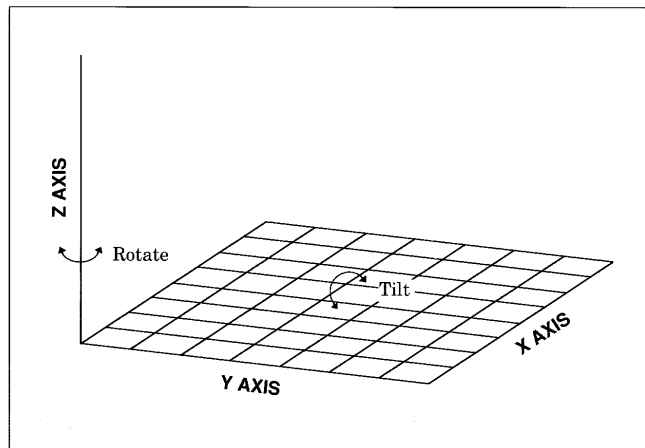
## About Rotating and Tilting the Plot

For both surface plots and scatter plots, you can rotate the  $x$ - $y$  plane about the  $z$  axis, or tilt the plot toward you. When you rotate a plot, you can view data from any angle around the three-dimensional graph. This is useful for bringing into view data points that were previously hidden by other data points on a plot. Tilting a plot enables you to accentuate the location of data points.

Figure 29.4 on page 979 shows how rotating and tilting can change the viewing angle of a graph.

*Note:* At certain combinations of tilt and rotation angles, the tick mark values may overlap.  $\Delta$

**Figure 29.4** Rotating and Tilting a Graph



---

## About Controlling the Axes

Because the relationship between a plot's surface and the actual data values can be difficult to interpret, you can improve a graph by changing the number of tick marks on the axes or restricting the range of the vertical ( $z$ ) variable.

The G3D procedure does not support AXIS definitions; however, you can use PLOT or SCATTER statement options to

- suppress the axes
- suppress axis labels
- suppress tick mark values
- specify the number of tick marks
- specify minimum and maximum values for the  $z$  axis
- specify whether grid lines connect axis tick marks.

You can also change the font and height of axis labels and axis values by specifying the desired font and height with the FTEXT= and HTEXT= options on a GOPTIONS statement.

For information on how to reverse the values on an axis, see "Reversing Values on an Axis" on page 992.

---

## Procedure Syntax

**Requirements:** At least one PLOT or SCATTER statement is required.

**Global statements:** FOOTNOTE, TITLE

**Reminder:** The procedure can include the BY, FORMAT, LABEL, NOTE, and WHERE statements.

**Supports:** Output Delivery System (ODS)

---

```
PROC G3D <DATA=input-data-set>
  <ANNOTATE=Annotate-data-set>
  <GOUT=< libref.>output-catalog>;
PLOT plot-request</options>;
SCATTER plot-request</option(s)>;
```

---

## PROC G3D Statement

Identifies the data set that contains the plot variables. Optionally specifies annotation and an output catalog.

**Requirements:** An input data set is required.

---

### Syntax

```
PROC G3D <DATA=input-data-set>
  <ANNOTATE=Annotate-data-set>
```

<GOUT=< *libref.*>*output-catalog*>;

## Options

**ANNOTATE=***Annotate-data-set*

**ANNO=***Annotate-data-set*

specifies a data set to annotate all of the graphs that are produced by the G3D procedure. To annotate individual graphs, use ANNOTATE= in the action statement.

**See also:** Chapter 10, “The Annotate Data Set,” on page 403

**DATA=***input-data-set*

specifies the SAS data set that contains the variables to plot. By default, the procedure uses the most recently created SAS data set.

**See also:** “SAS Data Sets” on page 25 “About the Input Data Set” on page 978

**GOUT=**< *libref.* >*output-catalog*

specifies the SAS catalog in which to save the graphics output that is produced by the G3D procedure. If you omit the *libref.*, SAS/GRAPH the catalog if it does not exist.

**See also:** “Storing Graphics Output in SAS Catalogs” on page 49

---

## PLOT Statement

Creates three-dimensional surface plots using values of three numeric variables from the input data set.

**Requirements:** Exactly one plot request is required.

**Global statements:** FOOTNOTE, TITLE

**Description** The PLOT statement specifies one plot request that identifies the three numeric variables to plot. This statement automatically

- scales the axes to include the maximum and minimum values for each of the plotted variables  $x$ ,  $y$ , and  $z$
- divides the value range for each variable into three even intervals, which are represented by four major tick marks on the axis
- rotates the  $x$ - $y$  plane  $70^\circ$  around the  $z$  axis and tilts it  $70^\circ$  toward you, labeling each axis with the name of the plotted variable or an associated label
- colors the plot with colors that are defined in the current colors list: axis labels and tick mark labels display in the first color from the list, axes display in the second color, the top of the surface plot displays in the third color, and the bottom of the surface plot (if visible) displays in the fourth color.

You can use statement options to modify any of the three plot axes as well as the general appearance of the graph, control the viewing angle, and specify characteristics for reference lines.

In addition, you can use global statements to add text to the graph, and an Annotate data set to enhance the plot.

### Syntax

**PLOT** *plot-request* </*option(s)*>;

*plot-request* must be

$y^*x=z$

*option(s)* can be one or more options from any or all of the following categories:

- appearance options:
  - ANNOTATE=*Annotate-data-set*
  - CBOTTOM=*bottom-surface-color*
  - CTOP=*top-surface-color*
  - ROTATE=*angle-list*
  - SIDE
  - TILT=*angle-list*
  - XYTYPE=1 | 2 | 3
- axes options:
  - CAXIS=*axis-color*
  - CTEXT=*text-color*
  - GRID
  - NOAXIS | NOAXES
  - NOLABEL
  - XTICKNUM=*number-of-ticks*
  - YTICKNUM=*number-of-ticks*
  - ZMAX=*max-value*
  - ZMIN=*min-value*
  - ZTICKNUM=*number-of-ticks*
- catalog entry description options:
  - DESCRIPTION=*'entry-description'*
  - NAME=*'entry-name'*

## Required Arguments

$y^*x=z$

specifies three numeric variables from the input data set:

*y*

is one of the variables that is plotted on the horizontal (*x-y*) plane.

*x*

is another of the variables that is plotted on the horizontal (*x-y*) plane.

*z*

is the variable that is plotted on the vertical (*z*) axis.

## Options

Options in a PLOT statement affect all graphs that are produced by that statement. You can specify as many options as you want and list them in any order.

**ANNOTATE**=*Annotate-data-set*

**ANNO**=*Annotate-data-set*

specifies a data set to annotate plots that are produced by the PLOT statement.

**See also:** Chapter 10, “The Annotate Data Set,” on page 403



**CAXIS=axis-color**

specifies a color for axis lines and tick marks. By default, axes are displayed in the second color in the current colors list.

**CBOTTOM=bottom-surface-color**

specifies a color for the bottom of the plot surface. By default, the bottom surface is displayed in the fourth color in the current colors list.

**Featured in:** Example 2 on page 996

**CTEXT=text-color**

specifies a color for all text on the axes, including tick mark values and axis labels. If you omit this option, a color specification is searched for in this order:

- 1 the CTEXT= option in a GOPTIONS statement
- 2 the default, the first color in the colors list.

**CTOP=top-surface-color**

specifies a color for the top of the plot surface. By default, the top surface is displayed in the third color in the current colors list.

**Featured in:** Example 2 on page 996

**DESCRIPTION='entry-description'****DES='entry-description'**

specifies the description of the catalog entry for the chart. The maximum length for *entry-description* is 240 characters. The description does not appear on the chart. By default, the procedure assigns a description of the form PLOT OF  $y^*x=z$ , where  $y^*x=z$  is the request that is specified in the PLOT statement.

**GRID**

draws reference lines at the major tick marks on all axes.

**Featured in:** Example 2 on page 996

**NAME='entry-name'**

specifies the name of the catalog entry for the graph. The maximum length for *entry-name* is 8 characters. The default name is G3D. If the specified name duplicates the name of an existing entry, SAS/GRAPH software adds a number to the duplicate name to create a unique entry, for example, G3D1.

**NOAXIS****NOAXES**

specifies that a plot have no axes, axis labels, or tick mark values.

**NOLABEL**

specifies that a plot have no axis labels or tick mark values. Use this option if you want to generate axis labels and tick mark values with an Annotate data set.

**ROTATE=angle-list**

specifies one or more angles at which to rotate the  $x$ - $y$  plane about the perpendicular  $z$  axis. The units for *angle-list* are degrees. By default, ROTATE=70. *Angle-list* is either an explicit list of values, or a starting and an ending value with an interval increment, or a combination of both forms:

$n$  <... $n$ >

$n$  TO  $n$  <BY increment>

$n$  <... $n$ > TO  $n$  <BY increment > < $n$  <... $n$ > >

The values specified in *angle-list* can be negative or positive and can be larger than 360°. For example, a rotation angle of 45° can also be expressed as

```
rotate=405
rotate=-315
```

You can specify a sequence of angles to produce separate graphs for each angle. The angles that are specified in the ROTATE= option are paired with any angles that are specified with the TILT= option. If one option contains fewer values than the other, the last value in the shorter list is paired with the remaining values in the longer list.

**See also:** TILT= option on page 984

**Featured in:** Example 2 on page 996

#### **SIDE**

produces a surface graph with a side wall.

**Featured in:** Example 3 on page 997

#### **TILT=*angle-list***

specifies one or more angles at which to tilt the graph toward you. The units for *angle-list* are degrees. By default, TILT=70. *Angle-list* is either an explicit list of values, or a starting and an ending value with an interval increment, or a combination of both forms:

```
n <...n>
n TO n <BY increment>
n <...n> TO n <BY increment > <n <...n> >
```

The values that are specified in *angle-list* must be 0 through 90.

You can specify a sequence of angles to produce separate graphs for each angle. The angles that are specified in the TILT= option are paired with any angles that are specified with the ROTATE= option. If one option contains fewer values than the other, the last value in the shorter list is paired with the remaining values in the longer list.

**See also:** ROTATE= option on page 983

**Featured in:** Example 3 on page 997

#### **XTICKNUM=*number-of-ticks***

#### **YTICKNUM=*number-of-ticks***

#### **ZTICKNUM=*number-of-ticks***

specify the number of major tick marks that are located on a plot's  $x$ ,  $y$ , or  $z$  axis, respectively. The value for  $n$  must be 2 or greater. By default, XTICKNUM=4, YTICKNUM=4, and ZTICKNUM=4.

**Featured in:** Example 2 on page 996

#### **XYTYPE=1 | 2 | 3**

specifies the direction of lines that are used to represent the surface. XYTYPE=1 displays the surface by using lines that represent  $y$  axis values. That is, it only draws lines that are parallel to the  $x$  axis. XYTYPE=2 displays the surface by using lines that represent  $x$  axis values, and draws only lines that are parallel to the  $y$  axis. XYTYPE=3 displays the surface by using lines that represent values for both the  $x$  and  $y$  axes, and creates a fishnet-like surface. By default, XYTYPE=3. See Figure 29.5 on page 985 for an example of the effect of XYTYPE= on the appearance of the surface.

#### **ZMAX=*max-value***

#### **ZMIN=*min-value***

specify the maximum and minimum values that are displayed on a plot's  $z$  axis. By default, the  $z$  axis is defined by the minimum and maximum  $z$  values that are in the

data set. You can use the ZMIN= and ZMAX= options to extend the  $z$  axis beyond this range. The value specified by ZMAX= must be greater than that specified by ZMIN=.

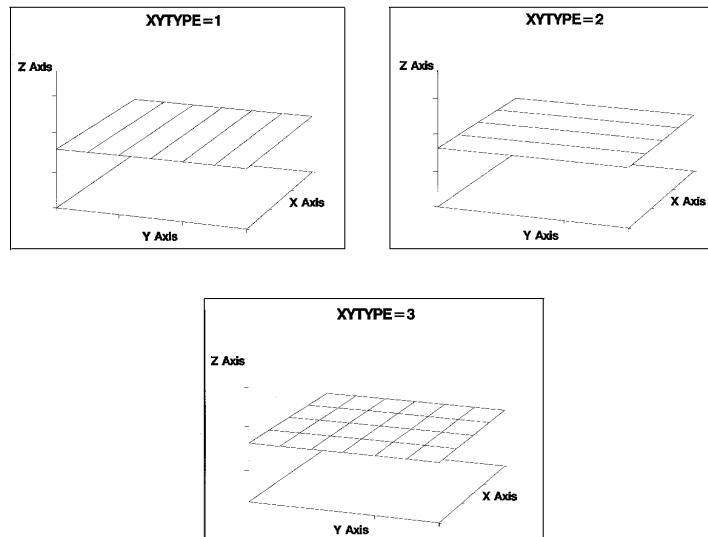
If you specify a ZMAX= or ZMIN= value within the actual range of the  $z$  variable values, the plot's data values are clipped at the specified level. For example, if the minimum  $z$  value in the data set is 0 and you specify ZMIN=1, the values of  $z$  that are less than 1 will be plotted as if they are 1.

**Featured in:** Example 2 on page 996

## Changing the Surface Appearance

Use the XYTYPE= option to change the appearance of the plot surface. This option lets you select the direction of the lines that form the surface plot. Figure 29.5 on page 985 shows examples of each type of plot surface.

Figure 29.5 Surface Appearance for Different XYTYPE= Values




---

## SCATTER Statement

Creates three-dimensional scatter plots using values of three numeric variables from the input data set.

**Requirements:** Exactly one plot request is required.

**Global statements:** FOOTNOTE, TITLE

**Alias:** SCAT

---

**Description** The SCATTER statement specifies one plot request that identifies the three numeric variables to plot. This statement automatically

- scales the axes to include the maximum and minimum values for each of the plotted variables  $x$ ,  $y$ , and  $z$

- divides the range for each variable into three even intervals that are represented by four major tick marks on the axis
- uses reference lines to mark the major tick marks on the  $x$  and  $y$  axes
- rotates the  $x$ - $y$  plane  $70^\circ$  around the  $z$  axis and tilts it  $70^\circ$  toward you, labeling each axis with the name of the plotted variable or an associated label
- colors the plot with colors that are defined in the current colors list: axis labels and tick mark labels display in the first color from the colors list, axes in the second color, and data points in the third color
- represents each data point with a pyramid that is connected to the horizontal plane with a needle.

You can use statement options to modify any of the three plot axes as well as the general appearance of the graph, control the viewing angle, and specify characteristics for reference lines. In addition, if the needles drawn from the data points to the base plane complicate a graph, you can suppress them.

You can use global statements to add text to the graph, and an Annotate data set to enhance the plot.

### Syntax

**SCATTER** *plot-request* </ *option(s)*>;

*plot-request* must be

$y*x=z$

*option(s)* can be one or more options from any or all of the following categories:

- appearance options:
  - ANNOTATE=*Annotate-data-set*
  - COLOR=*'data-point-color'* | *data-point-color-variable*
  - NONEEDLE
  - ROTATE=*angle-list*
  - SHAPE=*'symbol-name'* | *shape-variable*
  - SIZE=*symbol-size* | *size-variable*
  - TILT=*angle-list*
- axes options:
  - CAXIS=*axis-color*
  - CTEXT=*text-color*
  - GRID
  - NOAXIS | NOAXES
  - NOLABEL
  - XTICKNUM=*number-of-ticks*
  - YTICKNUM=*number-of-ticks*
  - ZMAX=*max-value*
  - ZMIN=*min-value*
  - ZTICKNUM=*number-of-ticks*
- catalog entry description options:
  - DESCRIPTION=*'entry-description'*
  - NAME=*'entry-name'*

## Required Arguments

***y\*x=z***

specifies three numeric variables from the input data set:

*y*

is one of the variables that is plotted on the horizontal (*x-y*) plane.

*x*

is another of the variables that is plotted on the horizontal (*x-y*) plane.

*z*

is the variable that is plotted on the vertical (*z*) axis.

The SCATTER statement does not require a full grid of observations for the horizontal variable.

## Options

Options in a SCATTER statement affect all graphs that are produced by that statement. You can specify as many options as you want and list them in any order.

**ANNOTATE=***Annotate-data-set*

**ANNO=***Annotate-data-set*

specifies a data set to annotate plots that are produced by the SCATTER statement.

**See also:** Chapter 10, “The Annotate Data Set,” on page 403

**CAXIS=***axis-color*

specifies a color for axis lines and tick marks. By default, axes display in the second color in the colors list.

**Featured in:** Example 6 on page 1003

**COLOR=***'data-point-color'* | *data-point-color-variable*

specifies a color name or a character variable in the input data set whose values are color names. These color values determine the color or colors of the shapes that represent a plot's data points. Color values must be valid color names for the device that is used. By default, plot shapes display in the third color in the current colors list.

If you specify **COLOR=***'data-point-color'*, all shapes are drawn in that color. For example, the procedure uses BLUE for all graph shapes when you specify

```
color='blue'
```

If you specify **COLOR=***data-point-color-variable*, the color of the symbol is determined by the value of the color variable for that observation. For example, the procedure uses the value of the variable CLASS as the color for each data point shape when you specify

```
color=class
```

Using **COLOR=***data-point-color-variable* enables you to assign different colors to the shapes to classify data.

**Featured in:** Example 5 on page 999

**CTEXT=***text-color*

specifies a color for all text on the axes, including tick mark values and axis labels. If you omit this option, a color specification is searched for in this order:

- 1 the CTEXT= option in a GOPTIONS statement
- 2 the default, the first color in the colors list.

**DESCRIPTION=***'entry-description'***DES=***'entry-description'*

specifies the description of the catalog entry for the chart. The maximum length for *entry-description* is 40 characters. The description does not appear on the chart. By default, the procedure assigns a description of the form SCATTER OF  $y^*x=z$ , where  $y^*x=z$  is the request that is specified in the SCATTER statement.

**GRID**

draws reference lines at the major tick marks on all axes.

**Featured in:** Example 5 on page 999

**NAME=***'entry-name'*

specifies the name of the catalog entry for the graph. The maximum length for *entry-name* is eight characters. The default name is G3D. If the specified name duplicates the name of an existing entry, SAS/GRAPH software adds a number to the duplicate name to create a unique entry, for example, G3D1.

**NOAXIS****NOAXES**

specifies that a plot have no axes, axis labels, or tick mark values.

**NOLABEL**

specifies that a plot have no axis labels or tick mark values. Use this option if you want to generate axis labels and tick mark values with an Annotate data set.

**NONEEDLE**

specifies that a plot have no lines that connect the shapes representing data points to the  $x$ - $y$  plane. The NONEEDLE option has no effect when SHAPE='PILLAR' or SHAPE='PRISM'.

**Featured in:** Example 5 on page 999

**ROTATE=***angle-list*

specifies one or more angles at which to rotate the  $x$ - $y$  plane about the perpendicular  $z$  axis. The units for *angle-list* are degrees. By default, ROTATE=70. *Angle-list* is either an explicit list of values, or a starting and an ending value with an interval increment, or a combination of both forms:

$n$  <... $n$ >

$n$  TO  $n$  <BY *increment*>

$n$  <... $n$ > TO  $n$  <BY *increment*> < $n$  <... $n$ > >

The values specified in *angle-list* can be negative or positive and can be larger than 360°. For example, a rotation angle of 45° can also be expressed

```
rotate=405
```

```
rotate=-315
```

You can specify a sequence of angles to produce separate graphs for each angle. The angles that are specified in the ROTATE= option are paired with any angles that are specified with the TILT= option. If one option contains fewer values than the other, the last value in the shorter list is paired with the remaining values in the longer list.

**See also:** TILT= option on page 990.

**Featured in:** Example 6 on page 1003

**SHAPE=***'symbol-name'* | *shape-variable*

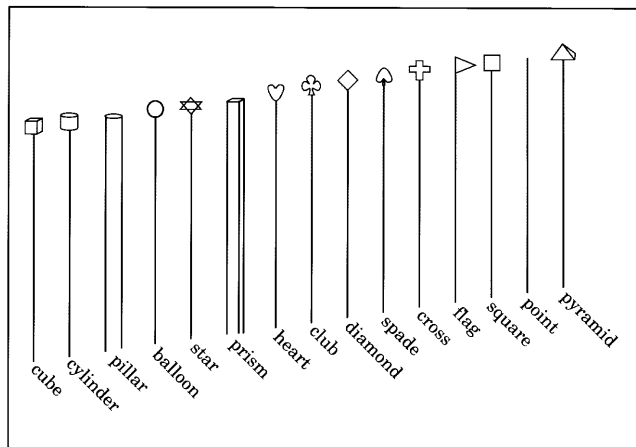
specifies a symbol name or a character variable whose values are symbol names. Symbols represent a scatter plot's data points. By default, SHAPE='PYRAMID'.

Values for *symbol-name* are

BALLOON  
 CLUB  
 CROSS  
 CUBE  
 CYLINDER  
 DIAMOND  
 FLAG  
 HEART  
 PILLAR  
 POINT  
 PRISM  
 PYRAMID  
 SPADE  
 SQUARE  
 STAR.

Figure 29.6 on page 989 illustrates these symbol types with needles.

Figure 29.6 Scatter Plot Symbols



If you specify `SHAPE='symbol-name'`, all data points are drawn in that shape. For example, the procedure draws all data points as balloons when you specify

```
shape='balloon'
```

If you specify `SHAPE=shape-variable`, the shape of the data point is determined by the value of the shape variable for that observation. For example, the procedure uses the value of the variable `CLASS` for a particular observation as the shape for that data point when you specify

```
shape=class
```

Using `SHAPE=shape-variable` enables you to assign different shapes to the data points to classify data.

**Featured in:** Example 5 on page 999

**SIZE=***symbol-size* | *size-variable*

specifies either a constant or a numeric variable, the values of which determine the size of symbol shapes on the scatter plot.

If you specify SIZE=*symbol-size*, all data points are drawn in that size. For example, if you specify SIZE=3, the procedure draws all symbol shapes three times the normal size. By default, SIZE=1.0. The units are in default symbol size.

If you specify SIZE=*size-variable*, the size of the data point is determined by the value of the size variable for that observation. For example, when you specify SIZE=CLASS, the procedure uses the value of the variable CLASS for each observation as the size of that data point. If you use SIZE=*size-variable*, you can assign different sizes to the data points to classify data.

**Featured in:** Example 6 on page 1003

**TILT=***angle-list*

specifies one or more angles at which to tilt the graph toward you. The units for *angle-list* are degrees. By default, TILT=70. *Angle-list* is either an explicit list of values, or a starting and an ending value with an interval increment, or a combination of both forms:

*n* <...*n*>

*n* TO *n* <BY *increment*>

*n* <...*n*> TO *n* <BY *increment*> <*n* <...*n*> >

The values that are specified in *angle-list* must be 0 through 90.

You can specify a sequence of angles to produce separate graphs for each angle. The angles that are specified in the TILT= option are paired with any angles that are specified with the ROTATE= option. If one option contains fewer values than the other, the last value in the shorter list is paired with the remaining values in the longer list.

**See also:** ROTATE= option on page 988

**XTICKNUM=***number-of-ticks*

**YTICKNUM=***number-of-ticks*

**ZTICKNUM=***number-of-ticks*

specify the number of major tick marks that are located on a plot's *x*{ *y*}, or *z* axis, respectively. The value for *n* must be 2 or greater. By default, XTICKNUM=4, YTICKNUM=4, and ZTICKNUM=4.

**Featured in:** Example 6 on page 1003

**ZMAX=***max-value*

**ZMIN=***min-value*

specify the maximum and minimum values that are displayed on a plot's *z* axis. By default, the *z* axis is defined by the minimum and maximum *z* values in the data. You can use the ZMIN= and ZMAX= options to extend the *z* axis beyond this range. The value that is specified by ZMAX= must be greater than that specified by ZMIN=. If you specify a ZMAX= or ZMIN= value within the actual range of the *z* variable values, the plot's data values are clipped at the specified level.

**Featured in:** Example 6 on page 1003

## Changing the Appearance of the Points

Use the COLOR=, SHAPE=, and SIZE= options to change the appearance of your scatter plot or to classify data using color, shape, size, or any combination of these features. Figure 29.6 on page 989 illustrates the shape names that you can specify in the SHAPE= option.



For example, to make all of the data points red balloons at twice the normal size, use

```
scatter y*x=z /color='red' shape='balloon' size=2;
```

To size your points according to the values of the variable TYPE in your input data set, use

```
scatter y*x=z / size=type;
```

For an example, see Example 5 on page 999.

## Simulating an Overlaid Scatter Plot

You can approximate an overlaid scatter plot by graphing multiple values for the vertical ( $z$ ) variables for a single ( $x, y$ ) position in a single scatter plot. To do this, add a small value to the value of one of the horizontal variables ( $x$  or  $y$ ) to give the observation a slightly different ( $x, y$ ) position. Thus, you enable the procedure to plot both values of the vertical ( $z$ ) variable. Represent each different vertical ( $z$ ) variable with a different symbol, size, or color. The resulting plot appears to be multiple plots overlaid on the same axes.

For example, suppose you want to graph a data set that contains two values for the vertical variable  $Z$  for each combination of variables  $X$  and  $Y$ . You could produce the original data set with a DATA step like this:

```
data planes;
  input x y z shape $;
  datalines;
1 1 1 PRISM
1 2 1 PRISM
1 3 1 PRISM
2 1 1 PRISM
2 2 1 PRISM
2 3 1 PRISM
3 1 1 PRISM
3 2 1 PRISM
3 3 1 PRISM
1 1 2 BALLOON
1 2 2 BALLOON
1 3 2 BALLOON
2 1 2 BALLOON
2 2 2 BALLOON
2 3 2 BALLOON
3 1 2 BALLOON
3 2 2 BALLOON
3 3 2 BALLOON
;
```

The SHAPE variable is assigned a different value for each different  $Z$  value for a single combination of  $X$  and  $Y$  values.

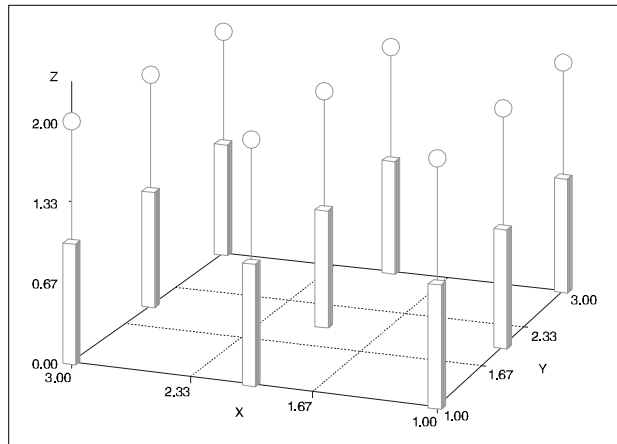
Ordinarily, the SCATTER statement only plots the  $Z$  value for the last observation for a single combination of  $X$  and  $Y$ . However, you can use a DATA step to assign a slightly different  $x, y$  position to all observations where  $Z$  is greater than 1:

```
data planes2;
  set planes;
  if z > 1 then x = x + .000001;
run;
```

Then you can use a SCATTER statement to produce a plot like the one in Figure 29.7 on page 992:

```
proc g3d data=planes2;
  scatter x*y=z / zmin=0 shape=shape;
run;
quit;
```

**Figure 29.7** Simulated Overlaid Scatter Plot



## Reversing Values on an Axis

Although you can use the SCATTER statement's ROTATE option to alter the view of a plot and therefore the general orientation to axes values, you cannot use SCATTER statement options to reverse axis values for one of the plot variables. To do this, you can multiply that variable's values by -1 to reverse the values themselves, which has the result of reversing the axis when those values are used to generate a plot. You should then use PROC FORMAT to define a format that displays the variable's values as they exist in the original data.

For example, the following code generates the scatter plot shown in Figure 29.8 on page 993:

```
data original;
  input y x z;
  datalines;
-1.15 1 .01
-1.00 2 .02
 1.20 3 .03
 1.25 4 .04
 1.50 5 .05
 2.10 1 .06
 2.15 2 .07
 2.20 3 .08
 2.25 4 .09
 2.30 5 .10
;
```

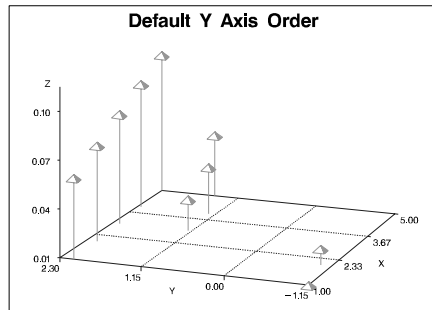
```

title1 'Default Y Axis Order';

/* default Y axis order */
proc g3d data=original;
  scatter y * x = z;
run;

```

Figure 29.8 Default Y-axis Order



To reverse the Y axis in the plot that is shown in Figure 29.8 on page 993, you can write a DATA step like the following to reverse the Y values and, therefore, reverse the Y axis when the values are plotted:

```

data minus_y;
  set original;
  y=-y;
run;

```

The previous code creates the MINUS\_Y data set by reading the ORIGINAL data set, and then multiplying the values of variable Y by -1. Although plotting Y values from the MINUS\_Y data set would reverse values on the Y axis, it would misrepresent the original data. Such a plot would label the axis with the negative-Y values. You can correct the problem by using PROC FORMAT to display Y values as they are stored in the ORIGINAL data set:

```

proc format;
  picture reverse
    low - < 0 = '09.00'
    0 < - high = '09.00' (prefix='-')
    0 = '09.00';
run;

```

Here, the PICTURE statement defines a picture format named REVERSE, which you can refer to in DATA and PROC steps by using the name followed by a period. A picture format is a template for printing numbers. The '09.00' specifications are *digit selectors* that indicate which digits or columns in the variable values will display in output; columns that do not have a specified digit selector will not be displayed in output. Thus, a picture format for displaying the values of variable Y needs a column for a minus sign, a column for units, and two columns for decimals. The digit selector 0 specifies that no leading zeros will display in a column, and the digit selector 9 specifies that a leading zero will display in a column.

The PICTURE statement defines this new picture format for three data ranges. The lowest value in the data up to but not including zero will display with no prefix, which

means negative values will display without a minus sign. All values above (but not including) zero to the highest value in the data will be displayed with the specified prefix, which in this case is a minus sign. Because zero is excluded from both ranges, it is assigned its own picture with no prefix.

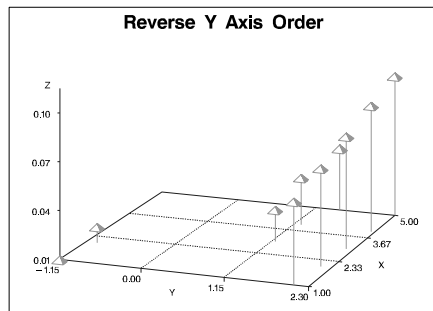
You can now assign the REVERSE format to the Y values from the MINUS\_Y data set and use Y to generate a scatter plot. The resulting plot displays Y's negative values without a prefix, and its positive values display with a minus sign prefix. This effectively represents Y values as they are stored internally in the ORIGINAL data set, thus correcting the data misrepresentation that results from multiplying Y by -1.

The following code generates the scatter plot shown in Figure 29.9 on page 994:

```
title1 'Reverse Y Axis Order';

/* reverses order of default Y axis */
proc g3d data=minus_y;
  format y reverse.;
  scatter y * x = z;
run;
quit;
```

**Figure 29.9** Reverse Y-axis Order




---

## Examples

---

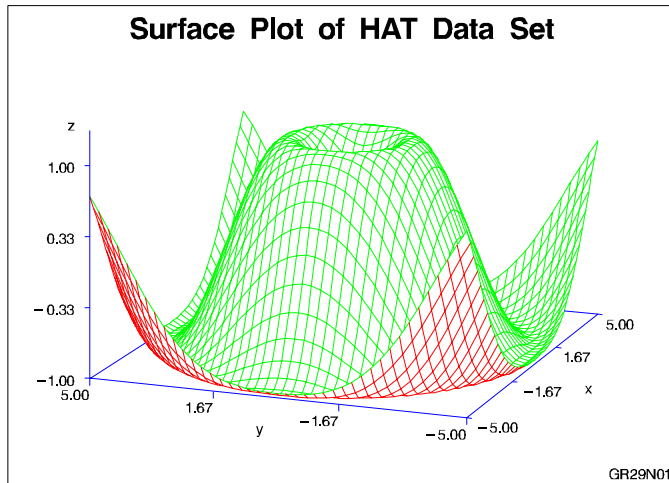
### Example 1: Generating a Default Surface Plot

Procedure features:

PLOT statement

Sample library member: GR29N01

---



This example shows a surface plot that reveals the shape of a generated data set named HAT. The PLOT statement in this example relies entirely on procedure defaults. The axes are scaled to include all data values and are labeled with the names of the axes variables. The axes major tick marks are divided into three even intervals, and the horizontal plane is rotated  $70^\circ$  around the  $z$  axis and tilted  $70^\circ$  toward you. The plot is displayed with the colors that the GOPTIONS statement defines for the colors list.

#### Assign the libref and set the graphics environment.

```
libname reflib 'SAS-data-library';
goptions reset=global gunit=pct border cback=white
         colors=(black blue green red)
         ftext=swiss ftitle=swissb htitle=6 htext=4;
```

**Create the data set.** REFLIB.HAT is generated data that produces a symmetric surface pattern, which is useful for illustrating the PLOT statement and its options.

```
data reflib.hat;
  do x=-5 to 5 by 0.25;
    do y=-5 to 5 by 0.25;
      z=sin(sqrt(x*x+y*y));
      output;
    end;
  end;
run;
```

#### Define title and footnote.

```
title 'Surface Plot of HAT Data Set';
footnote j=r 'GR29N01 ';
```

#### Generate the surface plot.

```
proc g3d data=reflib.hat;
  plot y*x=z;
```

```
run;
quit;
```

---

## Example 2: Rotating a Surface Plot

### Procedure Features

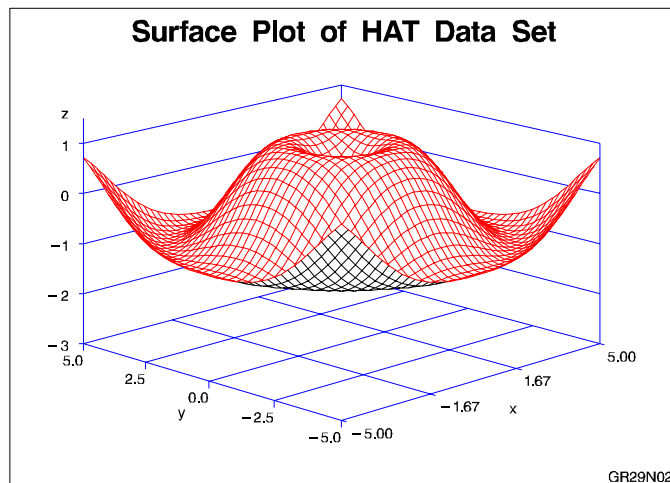
PLOT statement options:

```
CBOTTOM=
CTOP=
GRID
ROTATE=
YTICKNUM=
ZMAX=
ZMIN=
ZTICKNUM=
```

Data set: REFLIB.HAT on page 995

Sample library member: GR29N02

---



This example rotates the surface plot that is shown in Example 4 on page 998 and enhances its axes by adding reference lines and increasing the number of tick marks on the  $y$  and  $z$  axes. It also raises the plot above the horizontal  $x$ - $y$  plane.

### Assign the libref and set the graphics environment.

```
libname reflib 'SAS-data-library';
goptions reset=global gunit=pct border cback=white
          colors=(black blue green red)
          ftext=swiss ftitle=swissb htitle=6 htext=4;
```

**Define title and footnote.**

```
title 'Surface Plot of HAT Data Set';
footnote j=r 'GR29N02 ';
```

**Generate the surface plot.** GRID draws reference lines for all x, y, and z axis tick marks. ROTATE= specifies a rotation angle of 45°. CTOP= and CBOTTOM= change the colors of the plot's top and bottom surfaces. YTICKNUM= and ZTICKNUM= specify the number of tick marks for the y and z axes. ZMIN= and ZMAX= specify minimum and maximum values for the z axis. Specifying a minimum value that is below the minimum value in the data effectively raises the plot above the horizontal plane.

```
proc g3d data=reflib.hat;
  plot y*x=z / grid
      rotate=45
      ctop=red
      cbottom=black
      yticknum=5
      zticknum=5
      zmin=-3
      zmax=1;

run;
quit;
```

## Example 3: Tilting Surface Plot

Procedure features:

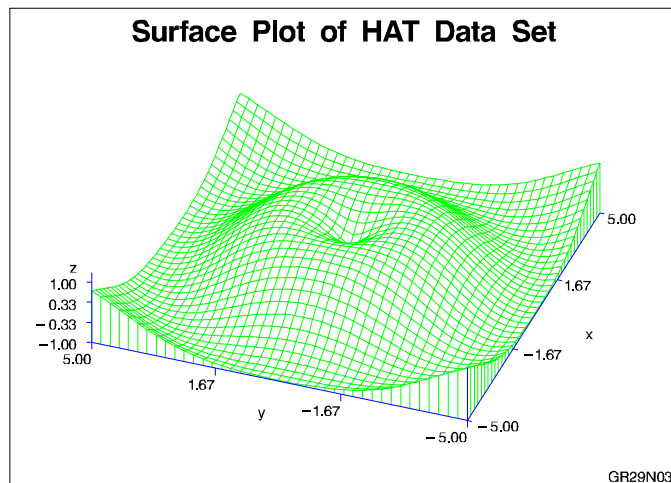
PLOT statement options:

SIDE

TILT=

Data set: REFLIB.HAT on page 995

Sample library member: GR29N03



This example modifies that shown in Example 1 on page 994 by tilting the surface plot  $15^\circ$  toward you and adding a side wall.

**Assign the libref and set the graphics environment.**

```
libname reflib 'SAS-data-library';
options reset=global gunit=pct border cback=white
        colors=(black blue green red)
        ftext=swiss ftitle=swissb htitle=6 htext=4;
```

**Define title and footnote.**

```
title 'Surface Plot of HAT Data Set';
footnote j=r 'GR29N03 ';
```

**Generate the surface plot.** SIDE draws a side wall for the graph. TILT= specifies a tilt angle of  $15^\circ$  for the plot, which doesn't affect the default rotation of  $70^\circ$ .

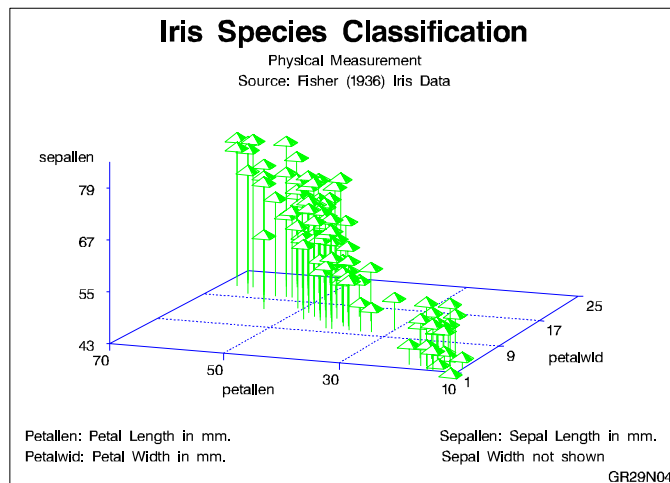
```
proc g3d data=reflib.hat;
  plot y*x=z / side
        tilt=15;
run;
quit;
```

## Example 4: Generating a Simple Scatter Plot

Procedure features:

SCATTER statement

Sample library member: GR29N04





This example shows a scatter plot that examines the results of measuring the petal length, petal width, and sepal length for the flowers of three species of iris. The SCATTER statement in this example relies entirely on procedure defaults, which scale the axes to include all data values, label the axes with the names of the axes variables, divide the axes into three even intervals, rotate the horizontal plane 70° around the  $z$  axis and tilt it 70° toward you, and display the plot with the colors that are defined for the colors list. The data points are represented by pyramids, which are connected to the horizontal plane with needles.

**Assign the libref and set the graphics environment.**

```
libname reflib 'SAS-data-library';
options reset=global gunit=pct border cback=white
        colors=(black blue green red)
        ftext=swiss ftitle=swissb htitle=6 htext=4;
```

**Create data set.** REFLIB.IRIS contains petal and sepal measurements for the flowers of three iris species, which are identified by species numbers.

```
data reflib.iris;
    input sepallen sepalwid petallen petalwid spec_no;
    datalines;
50 33 14 02 1
64 28 56 22 3
...more data lines...
63 33 60 25 3
53 37 15 02 1
;
```

**Define titles and footnotes.**

```
title1 'Iris Species Classification';
title2 'Physical Measurement';
title3 'Source: Fisher (1936) Iris Data';
footnote1 j=1 ' Petallen: Petal Length in mm.'
          j=r 'Sepallen: Sepal Length in mm. ';
footnote2 j=1 ' Petalwid: Petal Width in mm.'
          j=r 'Sepal Width not shown      ';
footnote3 j=r 'GR29N04 ';
```

**Generate a simple scatter plot.**

```
proc g3d data=reflib.iris;
    scatter petallen*petalwid=sepallen;
run;
quit;
```

---

## Example 5: Using Shapes in Scatter Plots

Procedure features:

SCATTER statement options:

COLOR=  
 GRID  
 NONEEDLE  
 SHAPE=

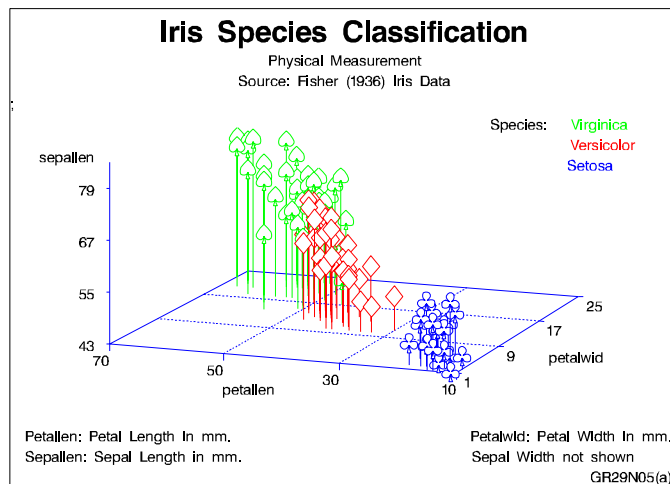
Other features:

DATA step  
 LABEL statement  
 NOTE statement

Data set: REFLIB.IRIS on page 999

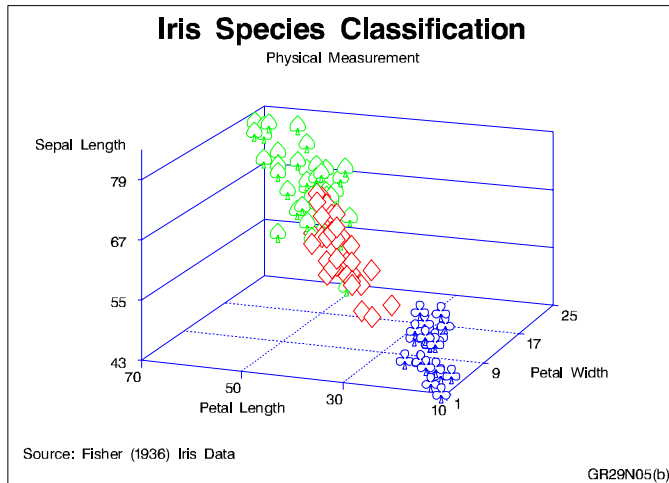
Sample library member: GR29N05

---



This program modifies that shown in Example 4 on page 998 to use shape symbols and color to distinguish information for various iris species. It also uses NOTE statements to simulate a plot legend.

The program then generates a second plot to modify the first. As shown by the following output, the second plot request suppresses the needles that connect data points to the horizontal plane, and adds reference lines to make it easier to interpret data values. It also labels the plot axes with descriptive text.



**Assign the libref and set the graphics environment.**

```
libname reflib 'SAS-data-library';
goptions reset=global gunit=pct border cback=white
          colors=(black blue green red)
          ftext=swiss ftitle=swissb htitle=6 htext=4;
```

**Create data set.** REFLIB.IRIS2 uses a DATA step to read and modify the REFLIB.IRIS data set. The DATA step adds a variable that identifies the iris species. It also adds two additional variables that store shape and color values for each iris species. These shapes and colors will distinguish iris species in the plot.

```
data reflib.iris2;
  set reflib.iris;
  length species $12. colorval $8. shapeval $8.;
  if spec_no=1 then
    do;
      species='setosa';
      shapeval='club';
      colorval='blue';
    end;
  if spec_no=2 then
    do;
      species='versicolor';
      shapeval='diamond';
      colorval='red';
    end;
  if spec_no=3 then
    do;
      species='virginica';
      shapeval='spade';
      colorval='green';
    end;
run;
```

**Define titles and footnotes.**

```

title1 'Iris Species Classification';
title2 'Physical Measurement';
title3 'Source: Fisher (1936) Iris Data';
footnote1 j=l ' Petallen: Petal Length in mm.'
           j=r 'Petalwid: Petal Width in mm. ';
footnote2 j=l ' Sepallen: Sepal Length in mm.'
           j=r 'Sepal Width not shown      ';
footnote3 j=r 'GR29N05(a) ';

```

**Generate the plot.** COLOR= specifies the variable that contains color information for the iris species. SHAPE= specifies the variable that contains shape information for the iris species.

```

proc g3d data=reflib.iris2;
  scatter petallen*petalwid=sepallen
         / color=colorval
           shape=shapeval;

```

**Create a legend using NOTE statements.** The first NOTE statement clears any existing notes. The second NOTE statement identifies the color key used for the different iris species.

```

note;
note j=r 'Species:      ' c=green 'Virginica      '
      j=r c=red 'Versicolor      '
      j=r c=blue 'Setosa      ';
run;

```

**Define new title and footnotes.**

```

title3;
footnote1 j=l ' Source: Fisher (1936) Iris Data';
footnote2 j=r 'GR29N05(b) ';

```

**Generate the plot.** NONEEDLE suppresses the line drawn from the x-y plane to the plot point. GRID draws reference lines for x, y, and z axis tick marks.

```

proc g3d data=reflib.iris2;
  scatter petallen*petalwid=sepallen
         / noneedle
           grid
           color=colorval
           shape=shapeval;

```

**Change the axes labels.** To improve axes labels, the LABEL statement associates labels with variable names.

```

label petallen='Petal Length'
      petalwid='Petal Width'

```

```

                sepallen='Sepal Length';
run;
quit;

```

## Example 6: Rotating a Scatter Plot

Procedure features:

SCATTER statement options

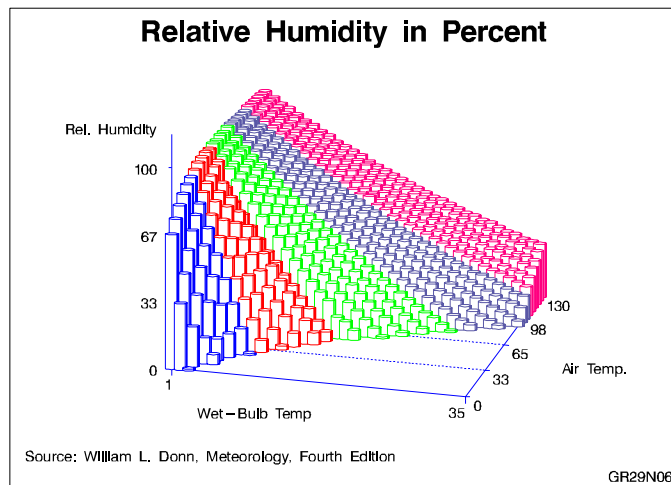
```

CAXIS=
ROTATE=
SIZE=
XTICKNUM
YTICKNUM=
ZMAX=
ZMIN=
ZTICKNUM=

```

Other features: DATA step

Sample library member: GR29N06



This example produces a scatter plot of humidity data. It uses color to distinguish air temperature ranges. The plot is rotated  $-15^\circ$ .

### Assign the libref and set the graphics environment.

```

libname reflib 'SAS-data-library';
goptions reset=global gunit=pct border cback=white
        colors=(black blue green red)
        ftext=swiss ftitle=swissb htitle=6 htext=4;

```

**Create data set REFLIB.HUMID.** The DATA step varies color according to specified air-temperature ranges.

```

data reflib.humid;
  length colorval $ 8.;
  label wtemp='Wet-Bulb Temp';
  label relhum='Rel. Humidity';
  label atemp='  Air Temp.';
  input atemp wtemp relhum;
  if atemp<26 then colorval="blue";
  else if atemp>=26 and atemp<+52 then colorval="red";
  else if atemp>=52 and atemp<+78 then colorval="green";
  else if atemp>=78 and atemp<+104 then colorval="lib";
  else if atemp>104 then colorval="pink  ";
  datalines;
0    1    67
0    2    33
...more data lines...
130 34    29
130 35    28
;

```

#### Define title and footnotes.

```

title 'Relative Humidity in Percent';
footnote1 j=1 ' Source: William L. Donn, Meteorology, Fourth Edition';
footnote2 j=r 'GR29N06 ';

```

**Generate the plot.** CAXIS= specifies a color for the axis lines and tick marks. ROTATE= specifies a rotation angle for the plot. SIZE= specifies the size of the plot symbols. XTICKNUM=, YTICKNUM=, and ZTICKNUM= specify the number of tick marks for the x, y, and z axes. ZMIN= and ZMAX= specify the minimum and maximum values for the z axis.

```

proc g3d data=reflib.humid;
  scatter atemp*wtemp=relhum
    / shape='pillar'
      color=colorval
      caxis=blue
      rotate=-15
      size=.5
      yticknum=5
      xticknum=2
      zticknum=4
      zmin=0
      zmax=100;

run;
quit;

```

---

## References

Fisher, R.A. (1936), "The Use of Multiple Measurements in Taxonomic Problems," *Annals of Eugenics*, 7, 179–188.

Watkins, S.L. (1974), "Algorithm 483, Masked Three-Dimensional Plot Program with Rotations (J6)," in *Collected Algorithms from ACM*, New York: Association for Computing Machinery.





The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS/GRAPH® Software: Reference, Version 8*, Cary, NC: SAS Institute Inc., 1999.

**SAS/GRAPH® Software: Reference, Version 8**

Copyright © 1999 by SAS Institute Inc., Cary, NC, USA.

ISBN 1-58025-525-6

All rights reserved. Printed in the United States of America.

**U.S. Government Restricted Rights Notice.** Use, duplication, or disclosure of the software by the government is subject to restrictions as set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, October 1999

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.

OS/2®, OS/390®, and IBM® are registered trademarks or trademarks of International Business Machines Corporation.

Other brand and product names are registered trademarks or trademarks of their respective companies.

The Institute is a private company devoted to the support and further development of its software and related services.