Spin, Rotate, and View Three Dimensional Point Clouds:

Dynamic Exploration of SAS Data Sets Without SAS/INSIGHT

Arthur L. Carpenter
Clinton A. Carpenter
California Occidental Consultants

KEY WORDS
AcroSpin, dynamic graphics, rotate, spin, point clouds

ABSTRACT
The graphical presentation of data is often critical to the investigator. This is especially the case during the data exploration phase of an analysis. Although SAS provides several tools for plotting two dimensions (PLOT, GPLOT, UNIVARIATE, etc.), the options are limited for the display of data using more than two dimensions (G3D). The real time exploration of three dimensional point clouds can be handled very nicely by using SAS/INSIGHT, however this software is not available to 6.04 or 6.06 SAS users.

An inexpensive non-SAS product, AcroSpin, is available for the PC, and provides much of the basic functionality of more expensive three-dimensional dynamic plotting packages. Although not difficult to use, the software requires a fairly strict structure for the data to be plotted, and for large or complicated data sets this can be a time consuming and tedious process.

The macro presented within this paper provides an interface between SAS and AcroSpin. Any data set with three numeric variables can easily be converted into the format used by AcroSpin. Once the conversion has taken place, AcroSpin is automatically executed, and the user is able to spin and rotate a three dimensional point cloud without ever leaving the SAS session.

ACROSPIN CONTROL FILES
AcroSpin has a specific format for its control files. There are two basic ways to present information in AcroSpin: as lines or as points. This paper focuses on the point presentation.

In the AcroSpin's code, each point must have five items of information. The needed information includes x, y, z coordinates, color, and layer. For lines, two endpoints must first be defined using coordinates and then named.

On a separate program line, the line is defined with the names of its two endpoints along with its layer and color. This information is presented in the following format.
It is easy to see how writing all of this information down for each point would be tedious. There are, however, a couple of options that reduce the size of the control file. The first shortcut is the set statement. The set statement defines any one of the five factors for all points following until redefined.

The second time saver is the pointlist or linelist statement. The pointlist statement defines the order of information of a point by eliminating the need to identify the numbers on each line. The following block of AcroSpin control code is a restatement of the above example using these short cuts.

```
set color 1
set layer 1
pointlist x y z name
0 0 0.47 a
5 0 0.48 xl
linelist from to
a xl
pointlist x y z layer color
5 5 -0.71 2 1
5 4.5 -0.43 2 1
```

By using a combination of color, layer, and the three coordinates, the relationships between five separate variables can be seen at once.

### INTERFACE MACRO

The points for the AcroSpin control file can be entered individually. Unfortunately, the time that this would take is restricting with even small data sets. In order to accurately and quickly transfer the data base into a AcroSpin file, a macro is needed. The macro presented here provides this interface.

The macro, MACSPIN, takes any three numeric variables and plots them as the three axes. In addition MACSPIN, allows for a fourth or even fifth variable to control the color and layer. The names of the appropriate SAS data step variables are passed to the MACSPIN macro as macro parameters. The SAS data set is then read and an ASCII AcroSpin control file is then created in a DATA _NULL_ step. The control file is then passed to the operating system through the use of the X command which executes AcroSpin. The interface is such that the user need not even leave the SAS session.

### USING ACROSPIN

AcroSpin is easy to learn to use. There are only a few commands required and the software comes with good documentation and a fun tutorial. The control keys required to rotate and spin the point cloud are shown below. This table is taken from the AcroSpin documentation and from Marsh, 1990.

<table>
<thead>
<tr>
<th>Arrow keys</th>
<th>move the point cloud</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC</td>
<td>exit from AcroSpin</td>
</tr>
<tr>
<td>R</td>
<td>rotate</td>
</tr>
<tr>
<td>Z</td>
<td>zoom</td>
</tr>
<tr>
<td>T</td>
<td>translate</td>
</tr>
<tr>
<td>L</td>
<td>toggle layers on and off</td>
</tr>
<tr>
<td>C</td>
<td>continuous rotation</td>
</tr>
<tr>
<td>H</td>
<td>on line help</td>
</tr>
</tbody>
</table>

### TIPS FOR EXPLORING DATA

The primary exploration tool and advantage of AcroSpin is dynamic three dimensional data plotting. Acrospin allows the user to dynamically view the data (point cloud) from any angle. The user is no longer constrained to lengthy time intervals between single frame snapshots of the data. When more than three numeric variables are available, the user must still select a subset of the available variables to be used in the plotting process, but the researcher is not restricted to only three variables in a given AcroSpin session.
Additional categorical discriminations can be specified by using colors and layering. Each data point may be assigned a color and a layer according to some user specified criteria. Then when plotted, the color may provide insight as to the relationships among the selected variables. Layering allows the user to view selected portions of the data independently of others. Layers may be displayed one at a time or in groups. Taken together, use of colors and layers is a powerful exploration tool.

Once a point cloud is plotted, AcroSpin allows the user full control of the viewing angle by dynamically spinning and rotating the point cloud at the touch of a key. Spinning the display often reveals relationships that might otherwise remain hidden. In addition to spin and rotation control, the data display may also be zoomed and translated. This allows the user to home in on specific portions of the data.

Often data exploration takes place in stages. Specific relationships are explored in more detail using a subset of the data. This is easily accomplished using SAS as the data management tool and with the MACSPIN macro as the interface with AcroSpin.

SUMMARY
AcroSpin is an inexpensive data exploration tool for an IBM/PC. Once interfaced with SAS, as through the use of the MACSPIN macro, this becomes a powerful and easy-to-use package that should be accessible to every statistician.

ABOUT THE AUTHORS
Arthur L. Carpenter has over fifteen years of experience as a statistician and data analyst and has served as a senior consultant with California Occidental Consultants, CALOXY, since 1983. His publications list includes a number of papers and posters presented at SUGI and he has developed and presented several courses and seminars on statistics and SAS programming.

Clinton A. Carpenter is a SAS and PRODAS programming consultant with California Occidental Consultants. Specialty projects include interfaces with SAS, including AcroSpin and CASE Tools. This is his second SUGI and his first SUGI presentation.

CALOXY offers statistical consultations, SAS contract programming, and in-house SAS training nationwide.
* macspin.sas
* create a three-dimensional point cloud using acrospin:
* written by: Arthur L. Carpenter & Clinton A. Carpenter
* CALOXY
* 4239 Serena Ave.
* Oceanside, CA 92056-5018
* (619) 724-8579
*
%macro macspin(dsn=,x=,y=,z=,level=,color=);
* call the macspin macro using the following named parameters:
* dsn - sas data set to rotate
* x - variable for x axis
* y - variable for y axis
* z - variable for z axis
* color - variable to designate colors
* 1 blue  9 intense blue
* 2 green 10 intense green
* 3 cyan  11 intense cyan
* 4 red   12 intense red
* 5 pink  13 intense pink
* 6 orange 14 intense orange
* 7 white 15 intense white
* 8 grey
* level - variable to designate levels (0 - 13000)
*;

* determine the median and maximum for each axis;
proc univariate data=&dsn noprint;
var &x &y &z;
output out=stat median=medx medy medz max=maxx maxy maxz;
run;

* create a flat file containing the AcroSpin commands;
data null;
merge &dsn (keep= &x &y &z &level &color) stat:
file '"&dsn..acd" noprint;
* the axes and axis labels are created first;
if n=1 then do;
  * axis length;
  ry = maxy-medy;
  rx = maxx-medx;
  rz = maxz-medz;
  maxr = max(ry, rx, rz);
  * letter size is a fraction of the max range size;
  * letter size increases with the size of the fraction;
  lu = .1*maxr;
  * determine critical point displacements;
  * the letter y lays in the xy plane;
  y2x = medx - lu;
  y3x = medx - lu/2;
  y4y = maxx - lu;
  y5y = maxx - lu/2;
  * the letter x lays in the xz plane;
  x2z = medz - lu;
  x3x = maxx - lu;
  x4x = maxr - lu;
  * the letter x lays in the yz plane;
  z2y = medy - lu;
  z3y = medy - lu/2;
  z4z = maxr - lu;
  put 'set color 13';
  put 'set layer 1';
  put 'endpointlist x y z name';
  * determine axis endpoints;
  put medx medy medz 'o';
  put maxx medy medz 'xl';
  put medx maxx medz 'yl';
  put medx medy maxr 'zl';
* determine the letter endpoints;
put y2x maxr medz 'y2';
put y3x y5y medz 'y3';
put y3x y4y medz 'y4';
put maxr medy x2z 'x2';
put x3x medy medz 'x3';
put x3x medy x2z 'x4';
put medx z2y maxr 'z2';
put medx medy z3z 'z3';
prompt medx medy z3z ‘z3’;
put medx z2y z3z 'z4';

* add lines;
put 'linelist from to';
* axes;
put 'o x1';
put '0 y1';
put 'o z1';
* letter x;
put 'x1 x4';
put 'x2 x3';
* letter y;
put 'y1 y3';
put 'y2 y3';
put 'y4 y3';
* letter z;
put 'z1 z2';
put 'z1 z4';
put 'z4 z3';

* designate points;
put 'pointlist x y z layer color';
end;
* write out the points to be plotted;
put &x &y &z &level &color;
run;
x "\acrospin\acrospin dsn..acd";
run;
%mend macspin;

* makehat.sas
* * make a cowboy hat and display using acrospin.
*;

* make a cowboy hat for acro spin;
data hat;
do x = -5 to 5 by .15;
do y = -5 to 5 by .15;
  z = sin(sqrt(x*x + y*y));
  if abs(x)>2.5 or abs(y)>2.5 or abs(z)>2.5 then lev=2 ;
  else lev=3;
  col = ceil(z*7)+8;
  if col > 15 then col = 15;
  output;
end;
run;
@include 'macspin.sas';
%macspin(dsn=hat, x=x, y=y, z=z, color=col, level=lev);
* iris.sas
* initial data and program based on the SAS sample library program
* DISCEX1 (DISCRIM DOCUMENTATION EXAMPLE 1) in SAS/STAT
* REF: FISHER (1936) IRIS DATA
*
proc format;
  value specname
    1 = 'SETOSA'
    2 = 'VERSICOLOR'
    3 = 'VIRGINICA';
  value specchar
    1 = 'S'
    2 = 'O'
    3 = 'V';
run:

data iris;
  title 'Discriminant Analysis of Fisher (1936) Iris Data';
  input sepal len sepalwid petallen petalwid species @;
  if species=1 then spcolor=11;
  else if species=2 then spcolor=13;
  else if species=3 then spcolor=15;
  label sepalallen='Sepal Length in mm.'
    sepalwid='Sepal Width in mm.'
    petallen='Petal Length in mm.'
    petalwid='Petal Width in mm.';
  cards;
  50 33 14 02 1 64 28 56 22 3
  65 28 51 15 3 46 34 14 03 1
  59 32 48 18 2 46 36 10 02 1
  65 30 52 20 3 56 25 39 11 2
  68 32 59 23 3 51 33 17 05 1
  77 38 67 22 3 63 33 47 16 2
  49 25 45 17 3 55 35 13 02 1
  50 33 14 02 1 60 29 45 15 2
  50 36 14 02 1 77 30 61 23 3
  57 29 42 13 2 72 30 58 15 3
  71 30 59 21 3 64 31 43 14 2
  49 24 33 10 2 56 21 29 11 2
  49 31 15 02 3 60 22 50 15 3
  66 29 46 17 3 77 39 14 2 60 34 45 16 2
  44 29 14 00 20 35 10 2 55 24 37 10 2
  47 32 13 00 1 46 31 15 02 1
  74 28 61 19 3 59 30 42 15 2
  56 28 49 13 2 60 22 40 10 2
  49 31 15 01 1 67 31 47 15 2
  56 30 41 13 2 63 25 49 15 2
  51 25 30 11 2 57 28 41 13 2
  54 39 13 04 1 51 35 14 03 1
  61 29 47 14 2 56 29 36 13 2
  68 30 56 21 3 55 25 40 13 2
  45 23 13 03 1 57 25 50 20 3
  55 23 40 13 2 66 30 44 14 2
  51 37 15 04 1 52 35 15 02 1
  63 33 60 25 3 53 37 15 02 1;
%include 'macspin.sas';
%macspin(dsn=iris, x=sepalallen, y=petallen, z=petalwid,
  color=spcolor, level=species);
%macspin(dsn=iris, x=sepalallen, y=sepalwid, z=petalwid,
  color=spcolor, level=species);
%macspin(dsn=iris, x=petallen, y=sepalwid, z=petalwid,
  color=spcolor, level=species);