Today there are a great number of environmental or geographical data sets that consist of matrices of numerical values. Some of these are matrices based on values aggregated into cells while others are regular arrays of sample points of continuous surfaces. The U.S. Geological Survey has produced a number of matrices of elevations for the U.S. and has a program to produce more over the years. The Digital Terrain Models or TOPCON Tapes as they are called are one-degree matrices of elevation covering the complete U.S. The elevations are spaced about 200 feet apart giving about 2 million values per matrix. There are now about 9,000 Digital Elevation Models based on 1:24,000 topographic quadrangle maps. In these models the elevations are sampled at 200 feet apart giving about 2 million values per matrix. Our Landsat satellites pass over everywhere in the world on an 18 day cycle, collecting numeric values in a grid of about 80 meter spacing. Great numbers of the digital matrices of Landsat data are now housed in universities and agencies. The National Oceanic and Atmospheric Administration, or NOAA, also has a number of satellites that receive numerical responses and these are made available into cells while others are regular arrays of sample points of continuous surfaces. The list of sample data and observational data in arrays goes on and on. There are also program packages such as SYMAP and SURFACE II that will take numerical data from an irregular array of points and interpolate the data to produce a regular array of values.

A number of turnkey systems and image processing systems are designed to work with these data sets, but such systems are not available to large numbers of people who have some need to work with or study any of these data sets. The SAS/GRAPH system offers some capabilities for handling, manipulating, and displaying this type of data and because of the widespread availability of this package it is important to examine what can be done with SAS/GRAPH procedures. The emphasis in this paper is on display, but the file manipulation capabilities in the SAS system and the statistical procedures to handle matrices give added reasons to consider the graphics capabilities.

Eyton (1983, 1176-77) told how he used SAS procedures for manipulating the large arrays and the statistical procedures PRINCOMP, FASTCLUS, and DTSMRM for the classification of selected scenes from Landsat data tapes. He argued for SAS procedures because they are well accepted by users and are readily available in many academic communities. He then went on to talk about how he graphically portrayed his matrices using three dimensional surface processing programs such as SYMVU. While we agree with Eyton that there are advantages in using SAS procedures for statistical analysis and file manipulation, we argue further that SAS/GRAPH procedures provide alternative methods of display of these data matrices, whether they be in raw form or after classification and manipulation. Three SAS/GRAPH procedures that we have used to display such matrices are G3D, GCONTOUR, and GPLOT.

Some Problems with GCONTOUR and G3D

The procedures GCONTOUR and G3D are designed to display numeric data such as these environmental matrices. But there are some limits to the direct use of these procedures. PROC GCONTOUR can be used to generate a contour plot with lines giving form to the surface at various levels (figure 1) or can be used with the PATTERN option to represent the shape of the surface with color/pattern combinations of small rectangles (figure 2). The contour lines are smooth and attractive, but without being able to label the lines it is difficult to visualize the shape of the surface. The PATTERN method offers improvements in this area, but smooth shapes end up with a "saw-tooth" effect which tends to misrepresent the smooth form of the surfaces. Also, the LEVEL command in figure 7 had to be set to the endpoint value of such level instead of the class break as in figure 1. It is very confusing when the same LEVEL values yield different cutoff points depending on the presence or absence of the PATTERN option. Some of the other problems with using GCONTOUR, and in some cases G3D, for the display of environmental surfaces include: 1) the inability of the procedure to portray non-rectangular study areas, 2) the dependence on the LEVEL option to set contour intervals at the exclusion of FORMAT Information which may have been provided, 3) the lack of capability to rotate the view or read the matrix from different orientations, 4) the inability to directly input rectangular matrices of data as many other programs can, and lastly, 5) the inability of the procedure to scale the X and Y axes proportionally in the plots. This last concern is perhaps the greatest problem. In GCONTOUR, as in many of the other SAS/GRAPH display procedures, the horizontal and vertical dimensions of the plot are scaled to fit the dimensions of the viewing window. In the procedure G3D the height of the surface is a function of the size of the viewing window and the angle of tilt and the user has no control over this dimension. If the titles and footnotes consume much of the height of the display screen or the plotter window, then the
resultant display will be compressed in the vertical dimension and expanded in the horizontal dimension. Examples of these proportionality problems are best shown in the SAS/GRAPH USER'S GUIDE. The GCONTOUR display of the Cowboy Hat on the cover of the manual is represented as a rectangle four times as wide as it is high, even though the original matrix is square and the labels on the axes show that each axis ranges from -5 to +5. In the manual, the G3D representations of the Cowboy Hat vary greatly in vertical exaggeration, depending on the tilt. For working with empirical data of the type discussed here, it is quite important to be able to scale the heights because vertical exaggeration is an important component in interpreting such surfaces.

We have found some ways to modify these proportionality problems. The easiest and most obvious is to change the width and height, which works in some situations, but to get exact proportionality is a hit or miss proposition. Another technique which can be used in GPL0T is to insert dummy Z-values at extreme X positions and then display those data points in the background color. Again, exact proportionality is a hit or miss game. The only true solution to this problem must come from the SAS/GRAPH programmers. Until this proportionality problem is solved, the effective use of SAS/GRAPH procedures for the display of environmental matrices will be limited.

Displaying Matrices with GPL0T

With the judicious use of procedure GPL0T it is possible to overcome many of the problems inherent with GCONTOUR, resulting in contour plots which are more aesthetically pleasing and easier to differentiate the levels of. Using the form PLOT Y=X along with a format for Z and matching PATTERN statements will produce this type of contour (figure 3). The use of GPL0T also overcomes the requirement of a rectangular X*Y surface because it is displaying the surface as a series of individual symbols. Since the visual impact of the display is a result of the many symbols forming the appropriate outline, the impact of this type of display is a function of the number of points being plotted. When dealing with dense representations of surfaces the display is very attractive, especially in color since the symbols start to overlap, and the color almost becomes solid fill. However, when dealing with less dense data sets, the distribution of points tends to be too sparse, detracting from the overall display.

To produce a more dense display of symbols representing the surface, the procedure G3GRID can be used. This procedure was designed to use interpolation techniques to generate a rectangular grid of data from a non-rectangular one, thus enabling the user to use GCONTOUR or G3D which requires a complete rectangular matrix as input. This procedure can also be used to take a sparse rectangular grid and interpolate points so that the resultant matrix representing a surface is more dense. Figure 4 shows the result of such manipulation. It is a display of the same data set and Z can be set to an arbitrary number not present in the actual data set. The user can then concatenate the outline data set with original data using the SET command and then add one more PATTERN statement. The PATTERN statement used in GPL0T is different from the PATTERN option used in GCONTOUR, and these should not be confused. With the PATTERN statement the user can use the interpolation command I=SPLINE to connect the points, or if the data set has fewer points, the I=SPLINE can be used to generate a smooth spline outline. When using I=JOIN, the last point in the outline must be the same as the first to complete the connection. However, when using I=SPLINE, not only must the first point be repeated at the end, but the first two points must be repeated to inform the spline routine where the curve is heading. Failure to do this will yield a smooth outline that has an abrupt straight line connecting the last point with the first. The example (figure 5) presents a "contour plot" with this type of outline. The plot has been kept extremely simple to make the effect of the outline more obvious. It can be seen that the outline began at the top of the plot, because the spline routine did not match completely as the outline finished. This double line would be hardly noticeable in a more realistic plot. This method can be extended to draw outlines between each level of Z if the user so desires. With a dense contour display, this extra work would probably add little to the final product.
Data Input and Rotation

Regardless of the method chosen to display a three dimensional surface, two problems face SAS users regarding data input and rotation. The data are often available in a rectangular array with Y rows and X columns and Z values at the intersection of each row and column. This presents no problem to other programs designed to deal with 3D surfaces, but the SAS system requires the data to be in three columns, for the variables X, Y, and Z. The data may also need to be rotated since CONTOUR and CPlOT have no rotation options, and G3D has the limited ability to rotate a surface through only 90 degrees.

To handle environmental matrices in the SAS/GRAPH packages, the senior author wrote a FORTRAN program to transform matrices from their original form into an appropriate input to the SAS system. In the process, provision was made to reorient the matrices so that any side may be at the bottom or the top. The junior author built upon this concept and wrote a SAS macro called MATRIX. The data can be read with an input statement such as:

```
INPUT (COL1 - COL100) (5.2);
```

The number of columns of data represent the number of X values in the matrix. The number of observations read in is the number of Y values. The actual values of the variables are the Z values at that XY combination. It is assumed that the first observation is the most northern border so that looking at the data set is analogous to looking at the top, east to the right, etc.

Under G3D, the USER'S GUIDE provides examples of how a mathematical surface was created in the DATA step. This matrix, or any matrix, in G3D can only be rotated through 90 degrees with the first row always in the front. This is adequate to view a symmetric function like the Cowboy Hat, but working with empirically derived data, it may be appropriate to view the matrix from any of 360 degrees.

Because empirically derived matrices do not have row and column values attached to them, a reasonable approach is to select the front of the matrix for viewing in the process of attaching the row and column values to the matrix. To some three dimensional plot programs it is easy to rotate through 360 degrees but the user has no frame of reference to ascertain where north is in the display, but with this program the user creates a new file which he/she knows will have the south-facing edge to the front or bottom, or whatever.

If the user would prefer a view with west at the top, he/she need only type:

```
%MATRIX (DATA=GRTD VIEW=WEST);
```

This command will invoke the macro listed in Appendix 1. The values of all symbolic parameters are the defaults, so it would not be necessary to supply them if the above values are applicable. The DATA parameter supplies the name of the input data set. The name of the output data set is LONG. The conventional COLn must be used to name the variables. The VIEW parameter has a default value of NORTH (i.e., no rotation). The macro invokes PROC MATRIX which in turn rotates the data 90 or 180 degrees depending on the view desired.

Upon completion of the rotation, or if no rotation was required, the macro prepares the data for use in plotting. The LONG data set will have three variables: X, Y, and Z. The number of observations in this data set will be equal to the number of Z values in the original matrix, that is the number of variables times the number of observations. This data set can then be used in any of the 3D procedures.

The MATRIX macro assumes that both X and Y values begin with one and increment by one. If, for example, X began with 50, Y began with 100, and both increased in units of ten, then all of the parameters in the MATRIX macro would be needed:

```
%MATRIX (DATA=GRTD VIEW=WEST FIRST X=50 DELTA X=10 FIRST Y=100 DELTA Y=10);
```

The results of this view would be identical to the previous example; only the scale would be changed. FIRST_X and FIRST_Y are the lowest values X and Y take on. DELTA_X and DELTA_Y are the units of change in each direction. However, that after rotation the FIRST_X and FIRST_Y would be reversed since the vertical axis is always considered to be Y.

The name of the macro and its other default values are also easily changed, but the names of the variables in the original data of the form COLn was chosen partially because this is in the default name PROC MATRIX which is used to do the rotation. It seemed easier to use this than to include a RENAME statement.

Conclusions

It is our hope that through the discussion of these problems and with the limited solutions that we offer others can find working with environmental data to numeric matrices to be more flexible and aesthetically pleasing. The SAS/GRAPH procedures contain many attributes that make them contenders for a basic system for the analysis and display of environmental data. The inability to retain the proportionalities of the original matrix strongly detracts from the use of most of the SAS/GRAPH procedures for professional displays, but the strength of the SAS/GRAPH system comes from its link with the complete SAS package which provides data handling and statistical procedures. In lieu of far more powerful turnkey graphics systems, scientists should look at the SAS and SAS/GRAPH products as viable alternatives to process and display parts of this ever growing supply of environmental data.

BIBLIOGRAPHY


265
APPENDIX

%MACRO MATRIX(DATAXGRID,VIEW=NORTH,FIRIPX=1,
DELTA_X=1,FIRST_Y=1,DELTA_Y=1);
PROC MATRIX; FETCH GRID DATA=DATA;
SHOWS=NROW(GRID);
NCOLS=NCOL(GRID);
GO TO VIEW;

*----------------------------------.-----------

NORTH:
*INITIATE ROTATED MATRIX WITH ZEROS;
ROTRATd=(1(NROWS,NCOLS,0));
*NO ROTATION IS NECESSARY FOR NORTHERN VIEW;
ROTRATd=GRID;
GO TO EXPAND;

*---------------------------------------------;

SOUTH:
*INITIALIZE ROTATED MATRIX WITH ZEROS;
ROTRATd=(1(NROWS,NCOLS,0));
*PLACE APPROPRIATE VALUES IN ROTATED MATRIX;
DO C=1 TO NCOLS;
DO R=1 TO NROWS;
ROTATED(R,C)=GRID(NROWS+1-R,NCOLS+1-C);
END;
END;
GO TO EXPAND;

*REDEFINE NROWS & NCOLS IF MATRIX IS NON-SQUARE;
IF NROWS NE NCOLS THEN DO;
NROWS=NROW(ROTRATd);
NCOLS=NCOL(ROTRATd);
END;

*---------------------------------------------;

EXPAND:
*INITIALIZE LONG DATA SET WITH THREE COLUMNS OF
ZEROS;
LONG=(1(NROWS*NCOLS,3,0));
*NOW EXPAND Y BY X MATRIX INTO A 3 BY (Y*X)
MATRIX;
ACCOUNT=1;
DO R=NROWS TO 1 BY -1;
DO C=1 TO NCOLS;
LONG(ACCOUNT,1)=(R-1)*DELTA_Y+(FIRST_Y);
LONG(ACCOUNT,2)=(C-1)*DELTA_X+(FIRST_X);
LONG(ACCOUNT,3)=(ROTATED(R,C),C);
ACCOUNT=ACCOUNT+1;
END;
END;
OUTPUT LONG OUT=LONG(RENAME=(COL1=Y COL2=X
COL3=Z));
%MEND MATRIX;
FIGURE 1. CONTOUR LINE PLOT USING CUTOFF LEVELS

FIGURE 2. CONTOUR PATTERN PLOT USING MIDPOINT LEVELS

FIGURE 3. GPLLOT Contour Before G3GRID
FIGURE 4. GPHOT CONTOUR WITH G3GRID OUTPUT

FIGURE 5. GPHOT "CONTOUR" WITH SPLINED OUTLINE