

Graphics for Statistical Concepts and Data Display

David A. Dickey, NC State University, Raleigh, NC

ABSTRACT

GRAPHICS CAN SERVE TO PRESENT RESULTS TO CLIENTS, TO EXPLAIN STATISTICAL PRINCIPLES, AND CAN ALSO REVEAL FEATURES IN A DATA SET THAT MAY NOT BE EVIDENT IN ROUTINE STATISTICAL ANALYSIS. THIS PAPER CONSISTS OF EXAMPLES IN WHICH I HAVE FOUND GRAPHICS TO BE HELPFUL. SOME OF THE PROGRAMS THAT CREATE THE GRAPHS ARE LONG SO RATHER THAN INCLUDING THEM, I PRESENT THE MOTIVATING SETTINGS, ANY PARTICULARLY RELEVANT CODE FEATURES, AND MOST IMPORTANTLY THE RESULTING GRAPHS. THE INTENT IS MORE TO MOTIVATE USING SAS/GRAPH® TO CREATIVE AND HELPFUL VISUAL MATERIAL THAN TO DISPLAY THE PARTICULAR SCRIPTS NEEDED TO ACCOMPLISH THE TASKS. THE INTENT IS TO MOTIVATE READERS TO USE THEIR OWN CREATIVE TALENTS TO IMPROVE PRESENTATIONS

INTRODUCTION

While one usually thinks of graphics as data presentation tools, I have used graphics both for that purpose and for displaying statistical concepts. At times rather complicated concepts become clearer when presented in graphical form. In this paper I present some graphs that I have found useful for both purposes. In some of the data analysis examples, the graphics revealed features of the data that were not evident in the statistical calculations. SAS code for some of the graphics can be found at <http://www4.stat.ncsu.edu/~dickey/SAScode/>

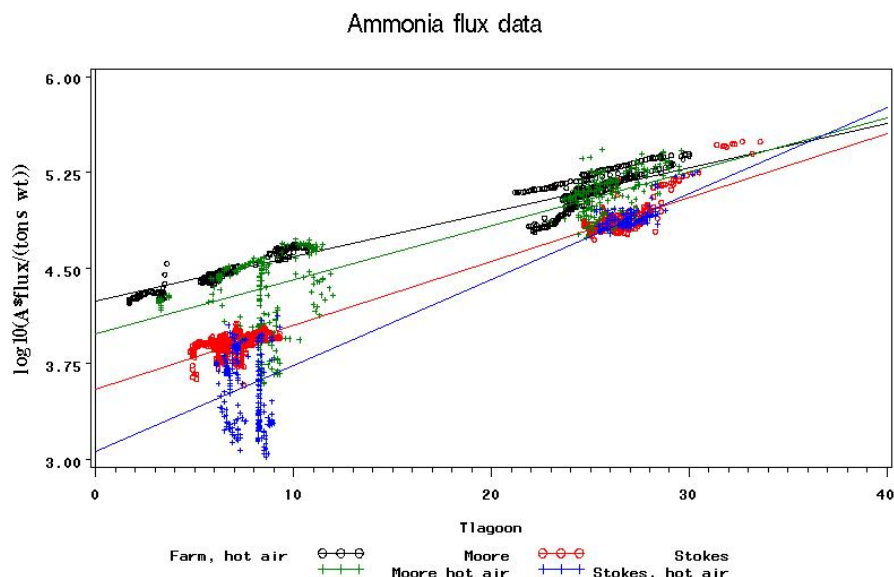
AMMONIA EMISSIONS EXAMPLE This first example is a case in which graphics gave some unexpected insights into a regression analysis. Hog farms have come under increased scrutiny in North Carolina because of the lagoons used to store waste, as well as other problems such as odor that go with the hog industry. Lagoon overflow during storms, of which North Carolina has many, is only one problem. Another is the emissions of gasses such as ammonia from the lagoon surfaces into the air. North Carolina is a major pork producing state. A moratorium was placed on new hog operations while researchers from NCSU, UNC-CH and Duke assessed some new technologies for reducing emissions and odors.

Part of this task was to measure ammonia flux on the lagoon surface. This was accomplished by floating a translucent plastic cylindrical chamber, 26 cm in diameter and 46 cm in height, fitted into a circular hole cut into the center of a floating plywood platform, which penetrated the lagoon surface by approximately 7 cm. Clean air was drawn into this chamber and out the opposite side then into a measuring device that showed how much ammonia had been added to the air. The physics of the lagoon/air interface suggested that the logarithm of this flux should have approximately a linear relationship with lagoon temperature. We begin with a plot of log flux (Y) versus temperature. Here I have created 4 groups passed on which of two farms was represented and whether the air temperature was higher than the lagoon temperature. The grouping variable is "color" with an associated format and these statements were used:

```
proc gplot data=ammonia; title H=1.5 "Ammonia flux data ";
  plot Lflux*TLagoon=color / haxis=axis1 vaxis=axis2 href=0;
axis1 major=(number=5);
axis2 major=(number=5) label=(angle=90 font=centb h=1.2);
symbol1 v=circle c=black h=0.5 i=rl;
symbol2 v=circle c=red h=0.5 i=rl;
symbol3 v=plus c=green h=0.5 i=rl;
symbol4 v=plus c=blue h=0.5 i=rl;
format color farm.;
```

The axis2 statement shows how to make the label run vertically and picks a font and height that seem to go well with the default graphics text. The h option makes the circles and plus signs smaller (there are many points) and the i=rl interpolates the points with linear regressions. The equations for these regressions are found in the log. The graph is in figure 1 .

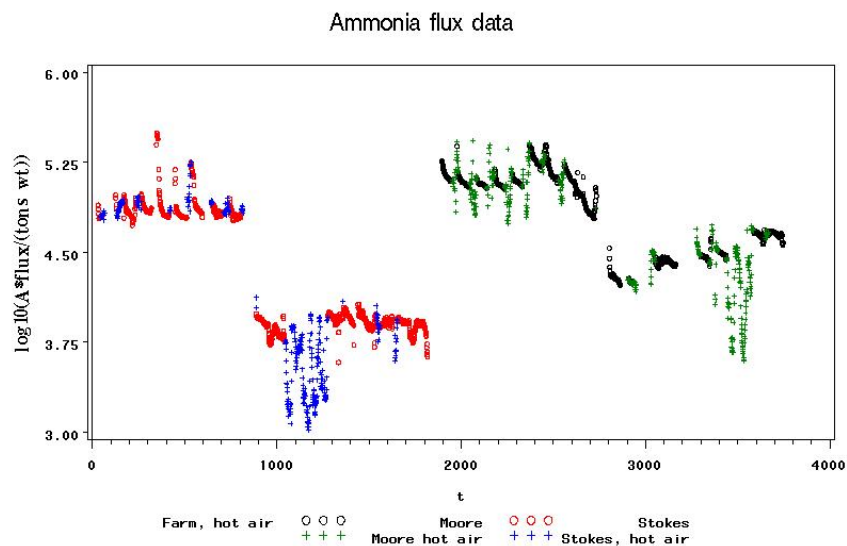
Figure 1. Ammonia Flux (log) Versus Lagoon Temperature



The two farms were measured in winter (left side) and in summer (right side of the graph). Clusters of plus signs hanging below the rest on the left side of the graph raised questions about simple linear regressions as reasonable models for the data.

Figure 2 shows the log(flux) values in the order they appear in the data set. We see the first farm in summer then winter on the left side then the second in summer and winter on the right. Again note the clusters of points, mostly + signs that hang below the rest, especially in the winter. From the legend you see the + signs indicate warmer air than lagoon temperatures so warm air seems to have a mitigating effect on emissions of ammonia as measured, that is, as the air temperature rises over that of the lagoon it seems to reduce our measurements of ammonia in the translucent cylinder, especially in winter. The same color scheme is used but with i=None replacing i=rl.

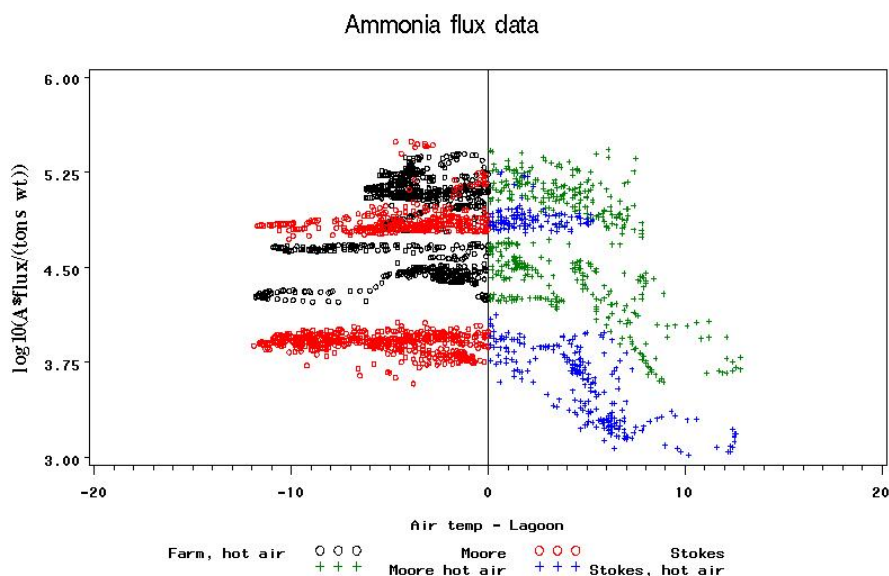
Figure 2. Emissions Versus Time



The researchers knew that lagoon temperature mattered, but we only discovered the air temperature effects by investigating the anomalies in the above graphs. Hindsight has been used in creating the graph symbols, in that the

coloring and symbols you see above were the result of investigating the inequality of variation apparent in the graphs. A graph of log(flux) versus the difference between air and lagoon temperature is quite revealing. Figure 3 shows this with the same color scheme but with i=none replacing i=rl.

Figure 3. Effects of Air – Lagoon Temperature Differences

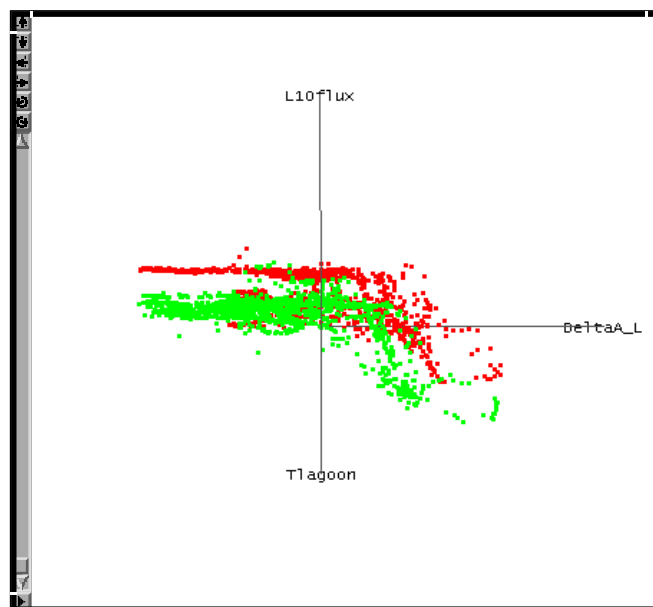


Points to the left of the vertical reference line (temperature difference 0) seem to fall into 4 horizontal groupings, corresponding to farm and season. On the right we have a similar grouping but the scattering of points appears to have a negative slope, roughly the same in both farms, at least in the winter data. A variable “hotair” was included along with lagoon temperature in the regression with the result that the R-square value was 95% and the residuals were more uniform. The hotair variable was the difference between air temperature and lagoon temperature except when that difference was negative, in which case hotair was set to 0.

In figure 3 you can see the effect of the difference between air and lagoon temperature but the effect of lagoon temperature itself, which is known to be important, is ignored. You could think of a lagoon temperature axis as coming straight out toward the viewer so the viewer is looking straight down the lagoon temperature axis. Figure 4 is a screen capture from SAS/INSIGHT, described below.

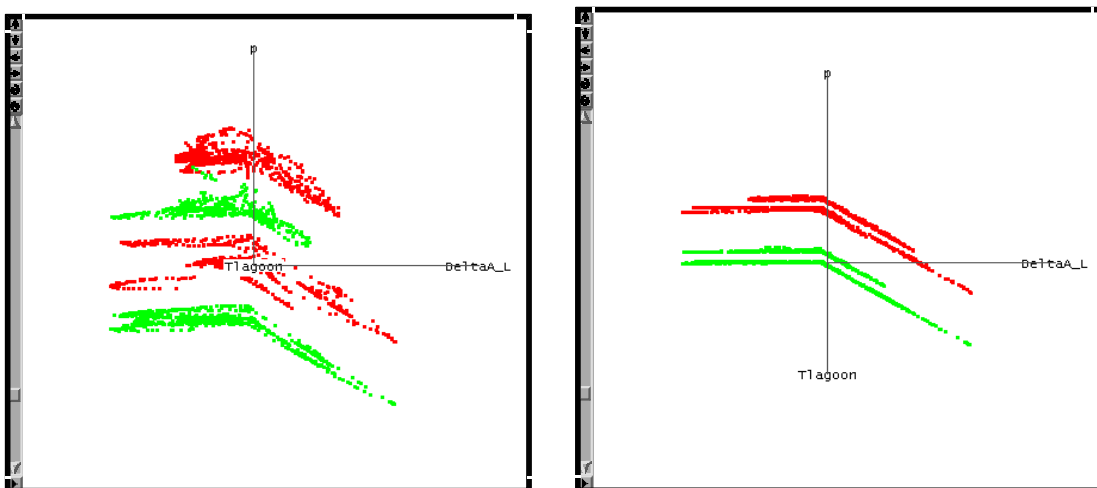
In INSIGHT, it is possible to make a rotating 3-D plot to capture the joint effect of the Air-lagoon difference and the lagoon temperature on ammonia flux. When properly rotated, it is seen that the fit is pretty tight and the error variation smaller than that of figure 3, the one ignoring lagoon temperature. A farm variable in the dataset was used to color the plot by farm. It is even clearer here that the “hotair” variable, which would allow a different slope on the right side of the screen capture graph than on the left side, is an important variable to fit. The tilting that it took to get the Tlagoon axis rotated into this position is the slope along that Tlagoon axis.

Figure 4. Rotating a Plot with INSIGHT.



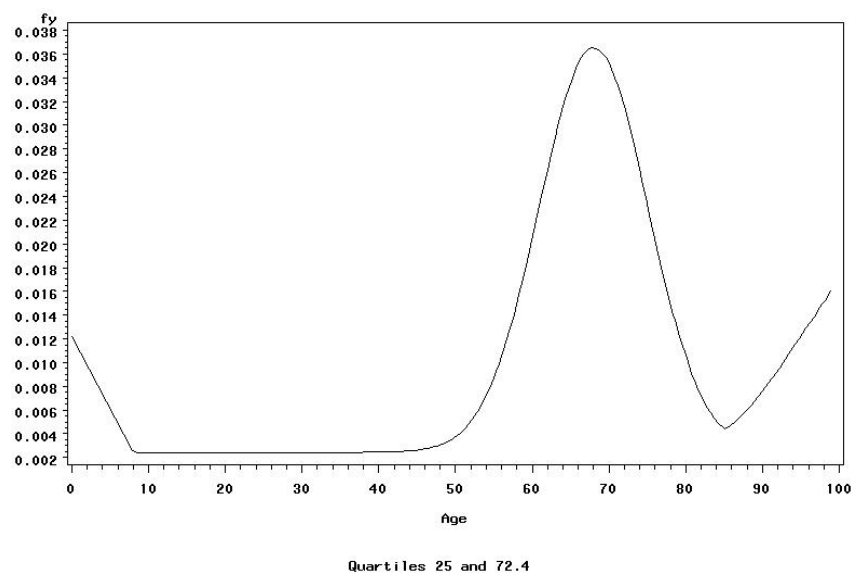
An idea of how well the model fit is given by plotting the predicted values in INSIGHT in the same way we did the data. On the left is the graph before tilting and on the right it is seen the proper tilt shows the segmented linear fit that was used.

Figure 5. Predicted Values Plotted as in Figure 3 and Rotated.



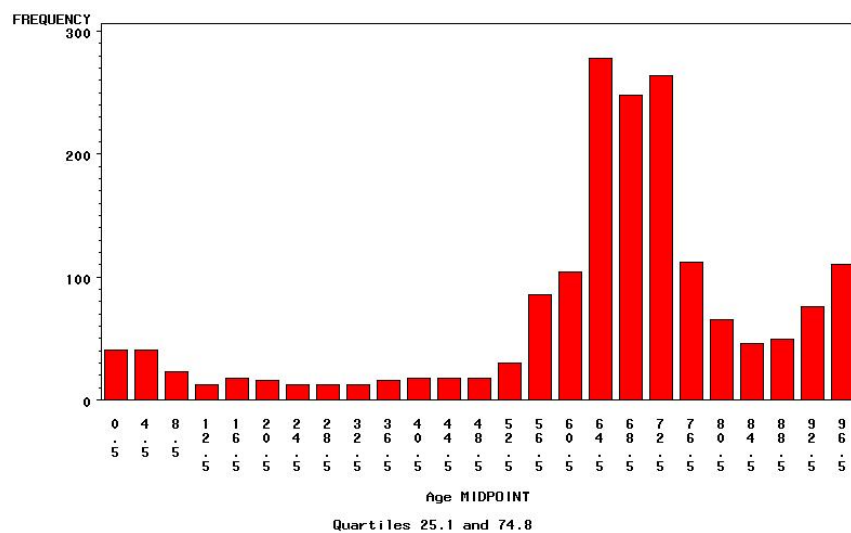
DEMOGRAPHIC DATA . A client asked how to find the shortest interval in a dataset with ages of death where the distribution seemed to be trimodal. The data here are made up to mimic that situation and come from this theoretical trimodal distribution:

Figure 6. Age at Death: Theoretical Density (assumed for illustration) .



The resulting data have the following histogram (PROC GCHART; VBAR AGE;)

Figure 7. Sample Histogram of Generated Age Data.



PROC MEANS gives the quartiles as 25 and 72.4 so that interval contains 50% of the data but is likely not the smallest interval that does so. Note the modes near 0, 66 and 96 as expected from the theoretical density. The trick is to sort these 1725 data points by age, compute $d = \text{age} - \text{lag863}(\text{age})$ and then sort to find the smallest d and print the results.

```
proc sort data=A; by age;
data B; set A;
high=age; low=lag863(age); D= high-low;
drop age;
proc sort data=B; by D;
proc print data=B (obs=5); where D ne .;
```

The output from this is as follows where id is the person's identification number:

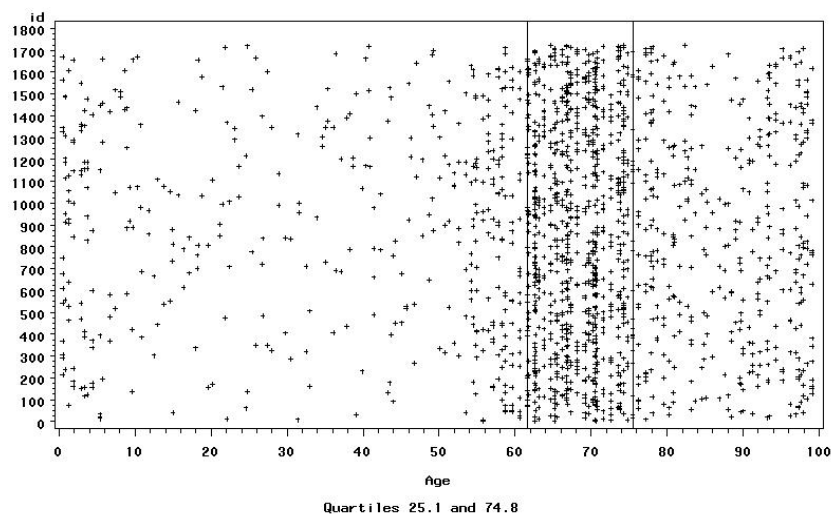
Obs	id	high	low	D
864	29	75.5	61.6	13.9
865	115	75.5	61.6	13.9
866	269	75.5	61.6	13.9
867	326	75.5	61.6	13.9
868	384	75.5	61.6	13.9

It is then a simple task to plot the data with reference lines at 61.6 and 75.5 to show the interval. I have used id as the vertical axis in the plot.

```
proc gplot data=A; plot id*age /href = 61.60 75.5;
symbol1 v=plus i=none h=0.5;
footnote "Quartiles 25.1 and 74.8";
```

to produce figure 7. This 50% containment interval is seen to be quite a bit narrower than that given by the quartiles: 27.4 versus 47.4. Had this been a simple normal distribution then, of course, the quartiles would have been the theoretical width minimizers.

Figure 8. Minimum Width 50% Containment Interval.

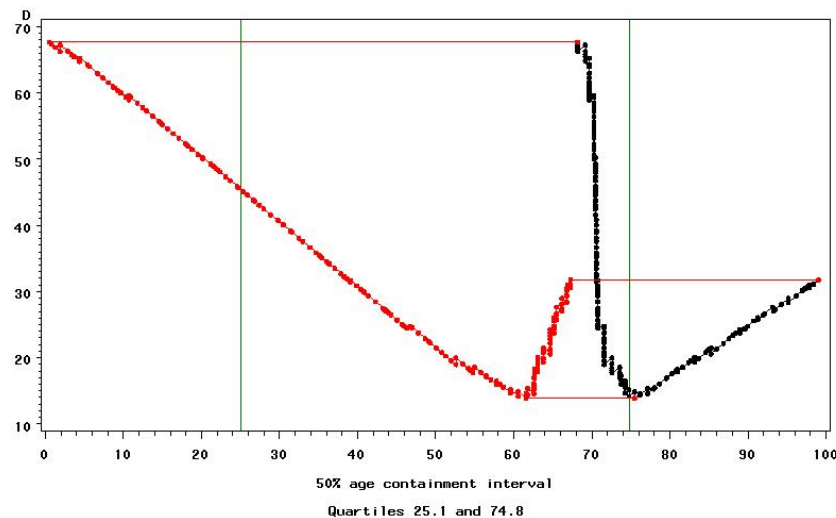


Another look at the intervals is given by

```
proc sort data=b; by low;
data b; set b end=eof;
if d>69 or d<27.8 or eof then do; save=low; low=high;
output; low=save; end; output;
proc gplot data=b; plot d*(high low)/overlay href=25 72.4 chref=green;
symbol1 v=dot i=join h=0.5 ;
label high="50% age containment interval";
run;
```

In the resulting graph (figure 9), points on the same horizontal line are endpoints of 50% containment intervals. These interval endpoints are joined together for cases where the width d exceeds 69 (at the top), cases near the minimum ($d < 27.8$) and the case at the end of the file. You can see how the intervals narrow and then widen again as the left interval endpoint increases and you can see how much narrower than the interquartile range is the width of the optimal interval.

Figure 9. All 50% Containment Intervals and their Widths (markers at quartiles).



FISH EXAMPLE. In a study of “fish recruitment,” data on catches of fish were available over a series of several years. The task was to investigate environmental features, both physical features and meteorological features, that seem to be associated with fish concentrations along the North Carolina shoreline. As a first cut, we wanted to see which study sites were coherent, that is, sites where increases and decreases in fish populations seemed to be similar over the years. The idea was to cluster the sites then look at characteristics of the sites within a cluster. Picking Spot as a study species, each site had a vector of yearly catches over a period of several years. Clustering on these sites (PROC CLUSTER) gave several clusters. Six of these clusters are shown below in figure 10, using PROC GPLOT to create the graphs. PROC GREPLAY is then used to place six of them in a template. An illustrative example of template creation is given in a later example. Note how the catch numbers rise and fall together over the years for the 16 years of data. For a good example, look at the 5 sites in the cluster in the right middle template area in figure 10 below. The upper left “cluster” is in fact a plot of those sites that had missing data and hence could not be clustered.

The map allowed our fellow researchers to think about geographic features that the sites in a cluster have in common. For example some clusters might be sites on the leeward side of coves etc.

The next step was to see how the clusters appeared on a map. Using the GMAP procedure with an annotate data set, it was possible to use dots to locate the sampled sites, coloring them by cluster. It was slightly difficult to see the clusters at a glance so I located a convenient point for each cluster from which rays of the appropriate color fanned out to the cluster members. The most important few clusters were shown on one graph and the rest on another. The resulting graph is figure 11. The light (yellow) lines in figure 11 point to sites with missing data. Figures 10 and 11 follow.

Figure 10. Fish Catch Clusters.

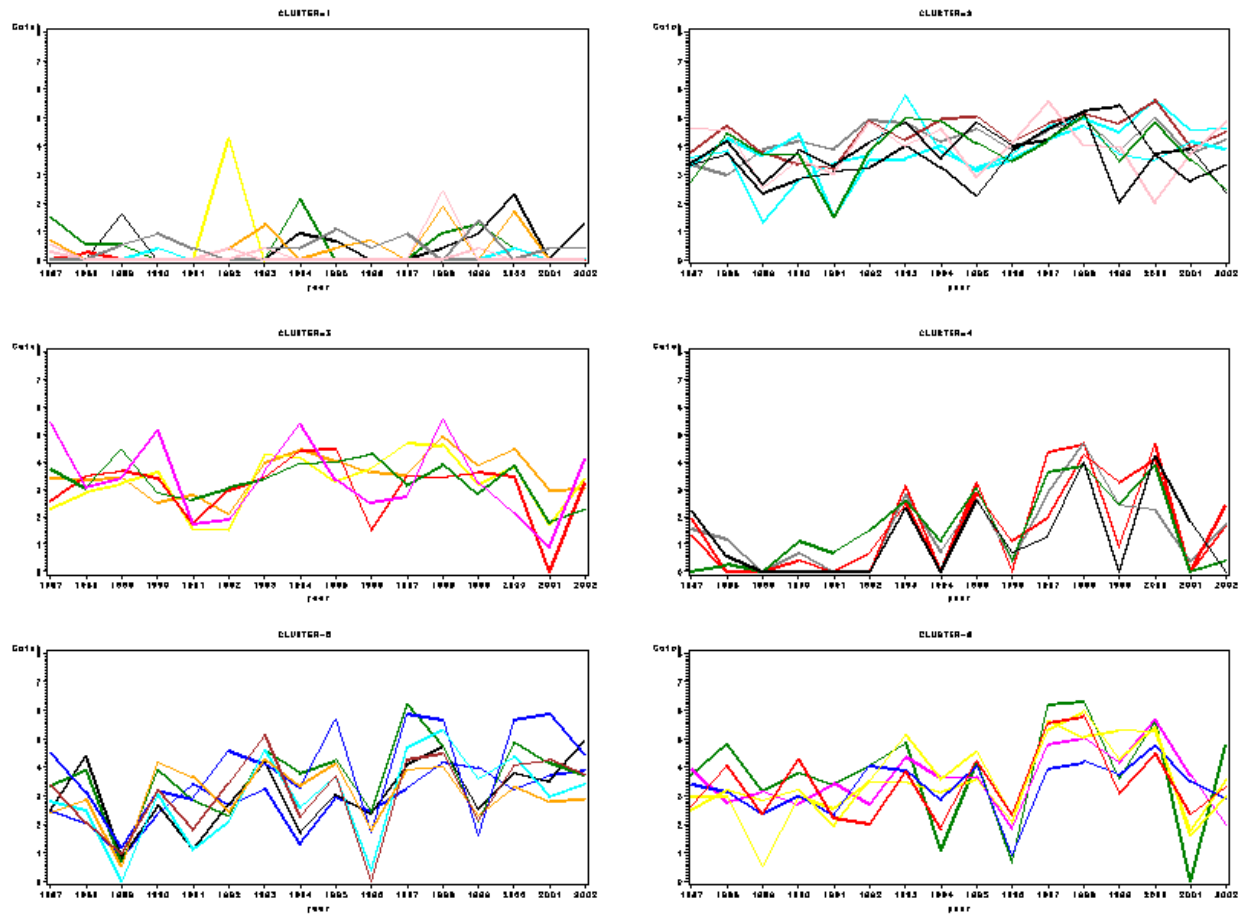
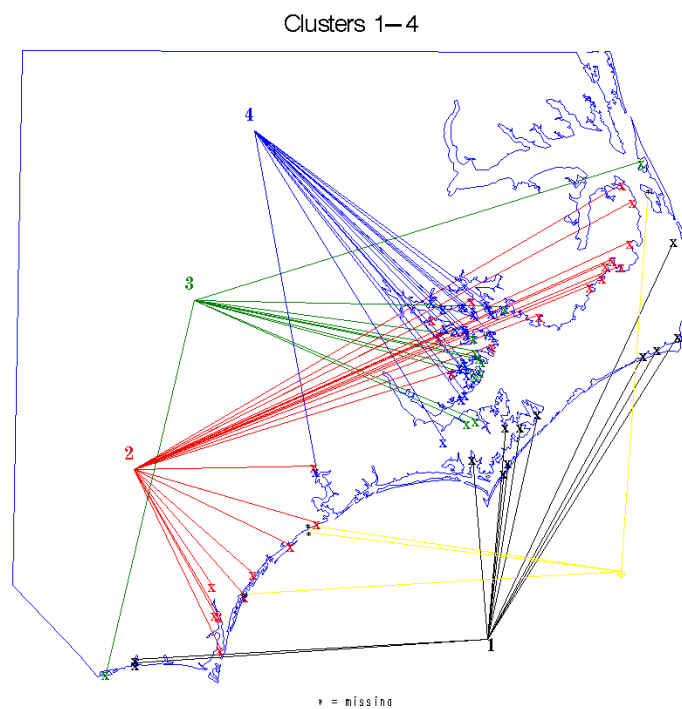
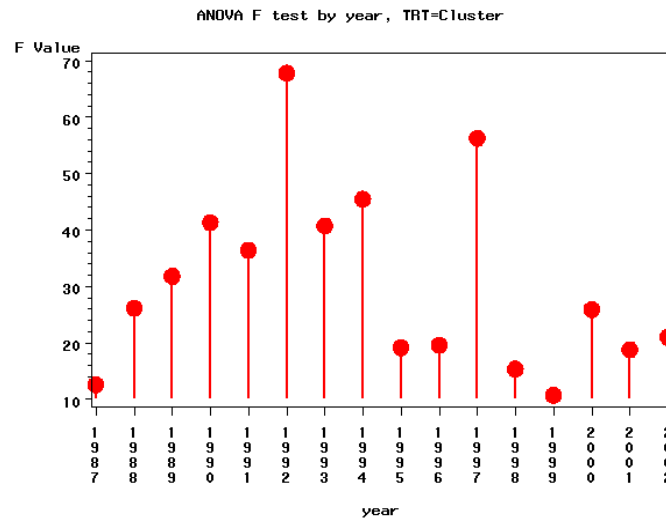


Figure 11. Mapping the Clusters.



Because we were interested in meteorological effects and because some clusters showed interesting variation over the years, the cluster mentioned above being an example, we looked for a method to see which years were most important in distinguishing the clusters. Each year there were catch numbers for all the sites, for example each x in figure 11 would be a point. Thinking of the clusters as treatments, we computed an analysis of variance F test so, for the four clusters illustrated in figure 11, this F would have $4-1 = 3$ numerator degrees of freedom. While these would not have F distributions (because the data clustering produced these “treatments” rather than them being determined beforehand) they do show the relative importance of different years. Plotting F for each year and using the v=dot and i=needle options in the symbol statement gave figure 12 in which years 1992 and 1997 appear to be particularly important in distinguishing the clusters from each other.

Figure 12. Pseudo F Tests Highlight Important Years.

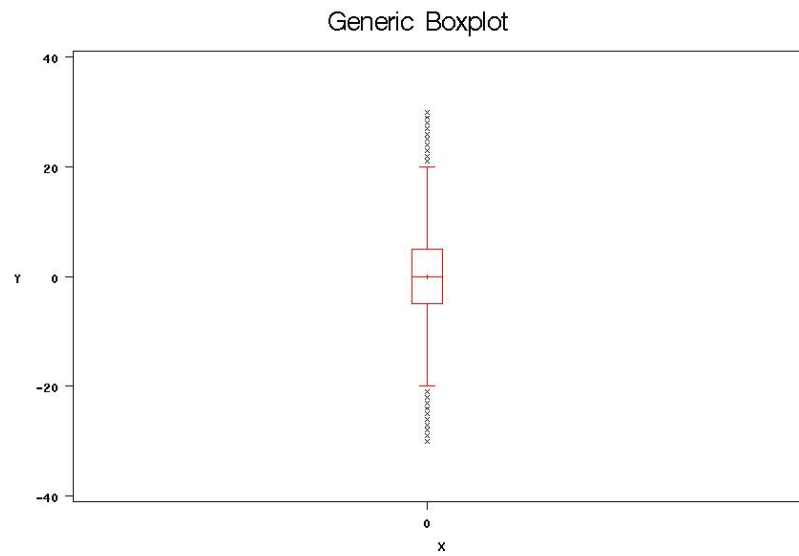


INVESTIGATING BOXPLOTS Another recent consulting experience was with a client who wanted to use boxplots to identify “outliers.” If we define an outlier to be something that does not arise from an assumed distribution, then the proportion of data falling outside the box “whiskers” will depend on what distribution we are looking at. First, we review boxplots.

The graph below is from data consisting of all the integers from -30 to 30 and 40 more zeros for 101 points altogether. The quartiles are -5, 0, and 5 so the box extends from $Q1=-5$ to $Q3=5$ for an interquartile range of 10. The whiskers on a box extend to the farthest points that are within 1.5 interquartile ranges of the box. The whisker ends are then at -20 and 20 so that 10 points are above the top whisker and 10 more below the bottom one. The procedure is designed to produce side by side boxplots for several X coordinates. A single $X=0$ is added to the data to produce the desired plot in figure 13.

```
Title "Generic Boxplot";
data box; X=0;
do Y = -30 to 30; output; end;
do i=1 to 40; Y=0; output; end;
drop i;
proc boxplot data=box;
plot Y*X / boxstyle=schematic
idsymbol=x idcolor=black idheight=0.5;
run;
```

Figure 13. An Illustrative Boxplot from PROC BOXPLOT.



The annotate facility creates a dataset with drawing instructions such as “move” and “draw.” A macro to create an annotate dataset that lets you place a boxplot horizontally below a graph using the annotate facility is shown here. The xsys and ysys values ‘2’ indicate that drawing will be with respect to the units on the graph’s axis. The arguments to the macro become macro variables so if “name” is replaced by “uni” then anno&name in the macro becomes announi, the name of an annotate dataset to annotate a graph.

```
%macro boxanno(name,ht,L1, q1, delta, q3, L2);
**
ht locates whisker axis->  --|---|
                        L1 q1 q3 L2
delta the half-width of the box and name names the
data set as anno&name ;

data anno&name; color = "red";
xsys = "2"; ysys="2";
function = "move"; Y=&ht;
X=&L1; output;
function="draw"; X=&q1; output;
Y=Y+&delta; output; X=&q3; output;
Y=Y-&delta; output; X= &L2; output;
X=&q3; output; Y=Y-&delta; output;
X=&q1; output; Y=&ht; output;
%mend boxanno;
```

First we look at how often points from a normal distribution would fall outside the whiskers. Consulting the normal tables (or using the PROBIT function) you find that -0.67 to 0.67 encloses the middle 50% of standard normal data so the box whiskers extend out to -2.68 and 2.68. Therefore the macro call below will set up a horizontal boxplot for the normal distribution at the Y coordinate -.1, with the plot lying in the horizontal strip lying between $y = -.13$ and $y = 0.07$ (because δ becomes -0.03). Also, again consulting the normal tables or using the PROBNORM function, you can find the probability $\Pr\{|Z| > 2.68\} = 0.007$, that is, theoretically only 7 in every 1000 points from a normal distribution are expected to lie outside the whiskers.

```
%boxanno(normal,-0.10, -2.68, -0.67, 0.03, 0.67, 2.68);
```

Here is the code to generate the normal curve and the annotated plot. For illustration, the w=2 symbol option is used to double the line's thickness.

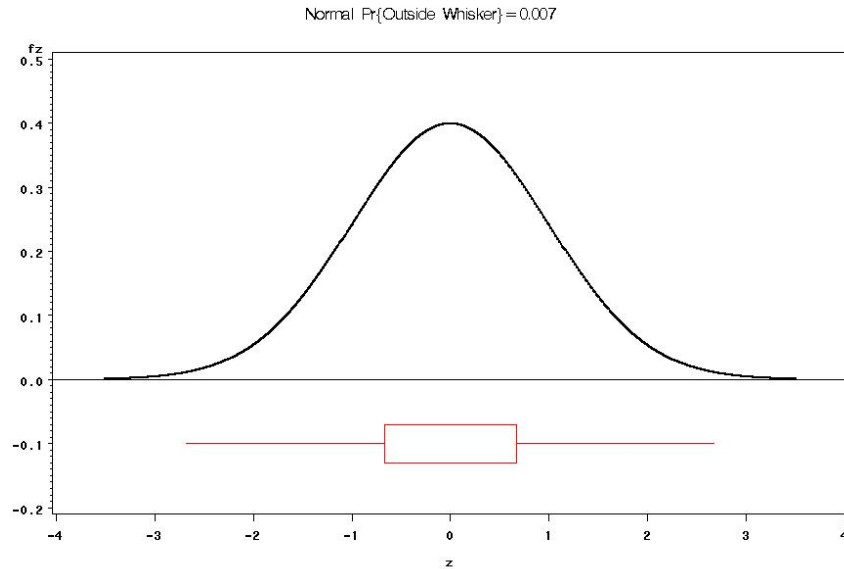
```

data graph;
do z = -3.5 to 3.5 by .01; z=round(z,.01);
fz = exp(-.5*z*z)/sqrt(2*3.14159);
output; end;
proc gplot data=graph annotate=annnonormal; plot fz*z/
vref=0 vaxis= -0.2 to 0.52 by .1;
symbol1 v=none i=join w=2;
title h=1.2 "Normal Pr{Outside Whisker}=0.007";

```

and here is the resulting plot, figure 14.

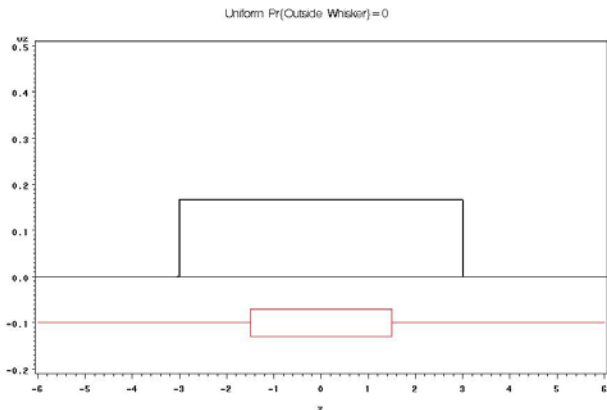
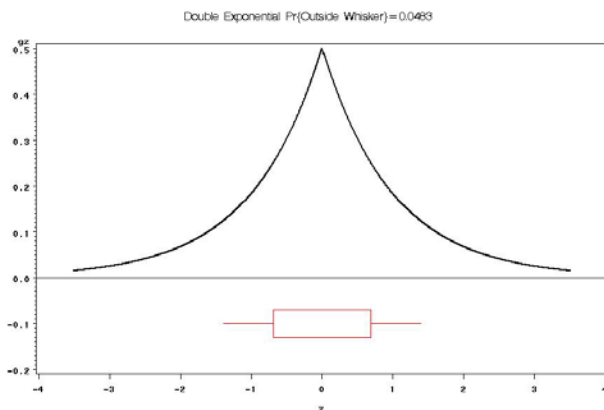
Figure 14. Annotated Normal Distribution.



Keep in mind that these are theoretical box plots. For finite samples, there would be variability in the dimensions of these. Theoretically, as we have seen from the probability 0.007 and can also see from the plot, it is unusual to have points falling outside the (theoretical) whiskers in a normal distribution. What happens with other distributions? One heavy tailed distribution is the double exponential and one with light tails (no tails you might say) is the uniform distribution. As with the normal, it is not too difficult to calculate the probabilities P of exceeding the whisker ends. These two comprise the figure 15.

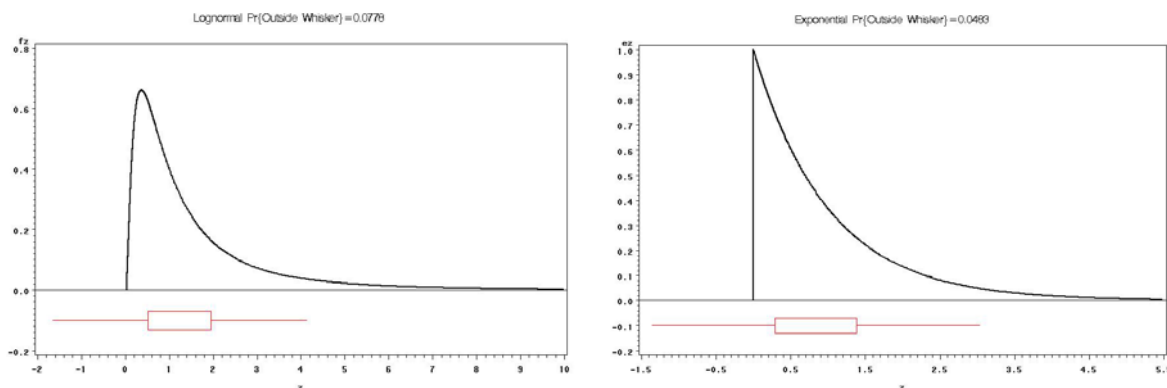
Figure 15 : (A) Double Exponential ($P=0.0483$),

(B) Uniform ($P=0$).



Another interesting thing to think about is the effect of skewness. Notice that in an asymmetric distribution there are still quartiles and an interquartile range. The theoretical whiskers are always equally long so the probability of being outside the whisker may be different on the high than on the low side. A very common highly skewed distribution is the lognormal. It is used often in studies of emissions of pollutants or amounts of contaminants in water. It is defined as $Y = \exp(X)$ where X is normal. Another is the single tailed exponential which is the appropriate distribution for waiting times between Poisson distributed events. Again with these distributions it is possible to calculate and insert into the macro the appropriate values and to compute the probability P of exceeding the whiskers. The lognormal is a bit interesting. The normal $Q1$ and $Q3$ would just be exponentiated to get the lognormal counterparts, however multiplying the resulting interquartile range by 1.5 and then computing the whisker ends results in two value not related in any simple way to the original normal whisker ends. Thus the areas beyond those whisker ends may not be the same on the high and low side. A glance at the resulting graphs in figure 16 illustrates this point:

Figure 16. (A) Lognormal ($P= 0.0778$) (B) Exponential ($P=0.0483$)



In summary, it is rather unusual to have observations outside the whiskers in a normal distribution, but in long tailed distributions, it can be not so rare. The lognormal, for example would have about 8% of its observations outside the whiskers. All of this, of course, begs the question "what is an outlier anyway?"

UNCORRELATED IS NOT THE SAME AS INDEPENDENT! In statistics classes we are taught that uncorrelated random variables are not necessarily independent. It is true that these two things are, in fact, equivalent in the normal distribution but not necessarily in others. Let Z be a random draw from a $N(0,1)$ distribution and let $Y = Z^2$. Thus Y has mean 1 and variance 2. Because Z is normal with mean 0, all of its odd moments are 0, that is $E(Z) = E(Z^3) = E(Z^5) = 0$ etc. where $E(\cdot)$ is the "expected value". With this in mind, you can see that $E(ZY) = 0$ so that with Z having mean 0, Z and Y are uncorrelated. However Z and Y can't possibly be independent in that if you know Z is 2, then Y has to be 4. To illustrate this idea, I will draw random samples of Z and compute a histogram of them and the associated Y values. I will make a "room" with one of the Z histogram on the back wall, the Y histogram on the left wall and a plot of Y versus Z in the floor.

The interesting part of this will be the creation of the "room" through the use of a template. Although most template applications use rectangles, the ability to specify all 4 template corners in SAS/GRAPH allows the user to make trapezoidal areas and this can be used to give depth perception. It is easier to see how this is done and what it produces by looking at the code and resulting plot than it is to describe it in words. These items follow.

First, because templates are to be used, code is submitted to clear the graphics catalog. The default in a SAS session is to accumulate graphs in the graphics catalog WORK.GSEG. Since the template entries consist of the graph names in that catalog, the template entries would have to be renumbered any time the program was rerun to correct a graph. The GREPLAY statement gets around this by clearing the catalog so that the default graph names are reused and the template instructions do not have to be updated on every run.

```
options reset=all;
proc greplay nofs; igout=work.gseg; delete _all_;
```

Next, generate the (Z,Y) points as described above and make two histograms and a quadratic plot.

```
data notindep;
do i=1 to 1000; Z=normal(1827655); Chisq=Z*Z;
```

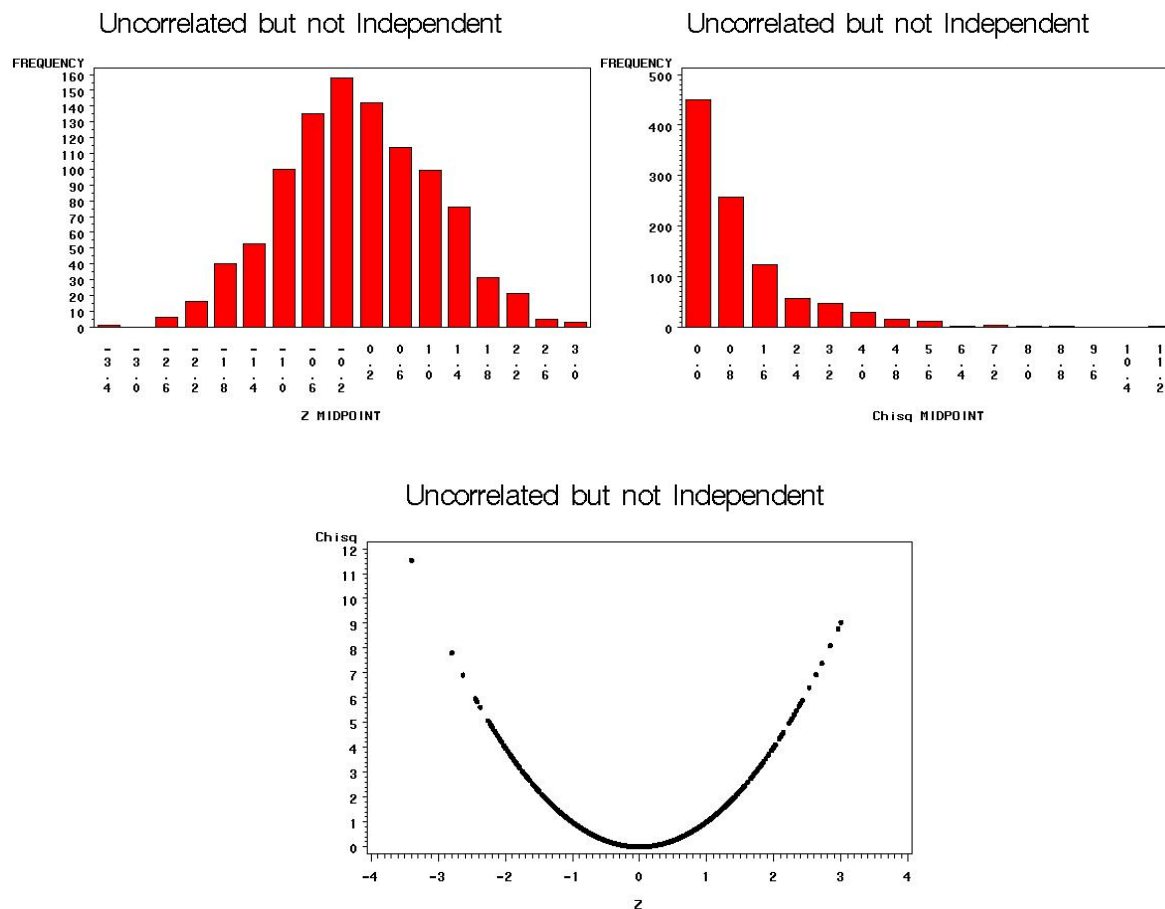
```

output; end; title "Uncorrelated but not Independent";
proc gchart; vbar Z Chisq;
proc corr; var Z Chisq;
proc gplot; plot Chisq*Z;
symbol1 v=dot h=.5;
run; quit;

```

These plots will come out in separate windows as usual and are shown as three panels in figure 17 .

Figure 17. Plots of Z and Y Histograms and of $Y = Z^2$.



Next, templates are created to put these together and achieve a three dimensional effect with depth perception. Artistic drawings achieve depth perception by using a vanishing point. For example to draw a cube with apparent depth, you could draw the square face on the viewer's side as a square, but then make the floor top and sides by drawing along lines connecting all four corners of the square to a single "vanishing" point, say in the upper right side of the graph. Rather than drawing all the way to that point the artist stops part way back and draws the back side square. Thus the front and back of the box are squares but the sides are in fact trapezoids.

Using this technique it is possible to create the back, floor, and left side wall of a box using trapezoidal template areas. Once done, this will allow us to display the three graphs above in a fashion that shows the relationships among them and brings the point of the demonstration clearly to the reader's view. The template statements follow. They are fairly intuitive with ulx indicating the l coordinate of the upper left corner, etc.

```

proc greplay tc=tempcat nofs igout=work.gseg;
tdef cube des='Cube'

1/ llx = 0  lly = 0
   ulx = 30 uly = 20
   urx = 100 ury = 20

```

```

lrx = 90 lry = 0

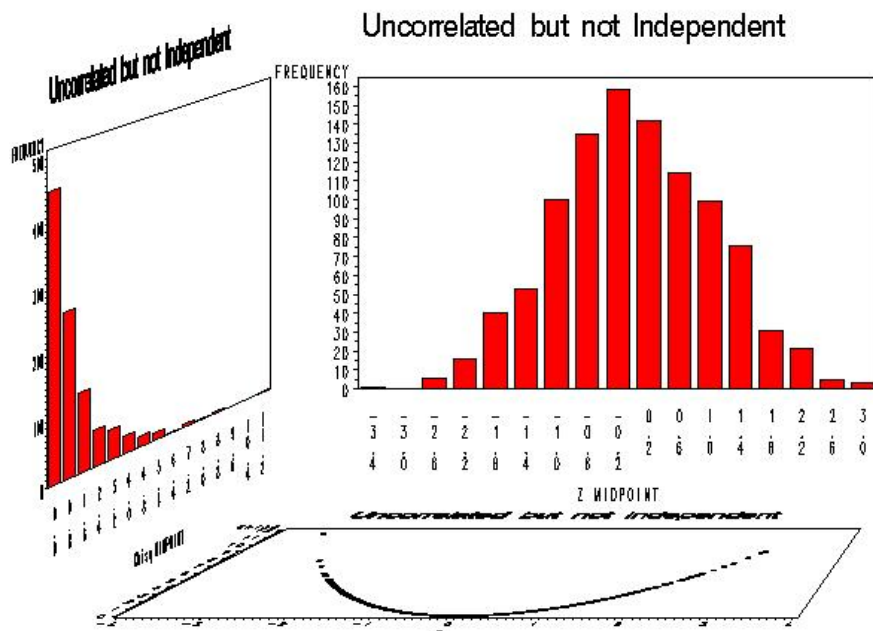
2//lx = 30 lly = 20
ulx = 30 uly = 100
urx = 100 ury = 100
lrx = 100 lry = 20

3//lx = 0 lly = 0
ulx = 0 uly = 88
urx = 30 ury = 100
lrx = 30 lry = 20
;
template = cube;
treplay 1:gplot 2:gchart 3:gchart1;
run;

```

The TREPLAY command tells the procedure which graphs to place in which part of the box. Because the template procedure calculates the deformation required for each graph, in other words it correctly rescales every point in the graph, the result as a three dimensional looking picture with apparent depth. This is figure 18.

Figure 18. A Cube Plot with Depth Perception.



RESIDUAL DIAGNOSTICS

The data here are made up to show a relationship between a response variable Y and two inputs x1 and x2. The data are first plotted using the following code.

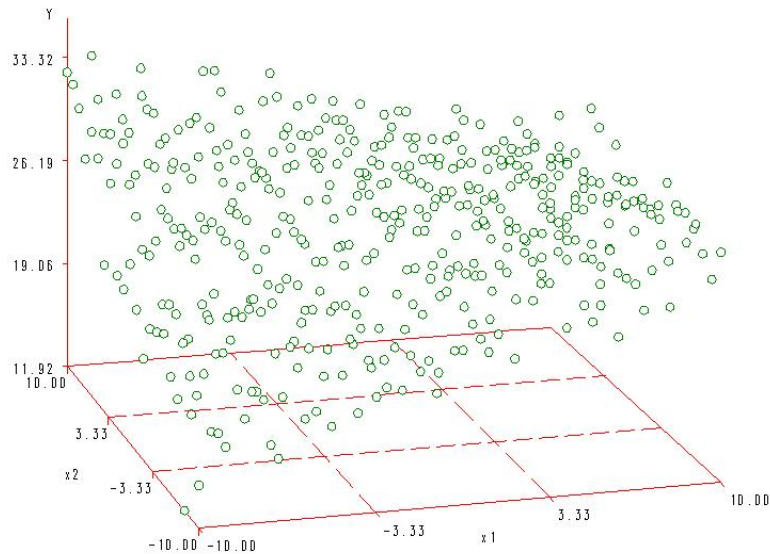
```

proc g3d;
scatter x2*x1=y / shape = "balloon" size=0.5 noneedle rotate=15;

```

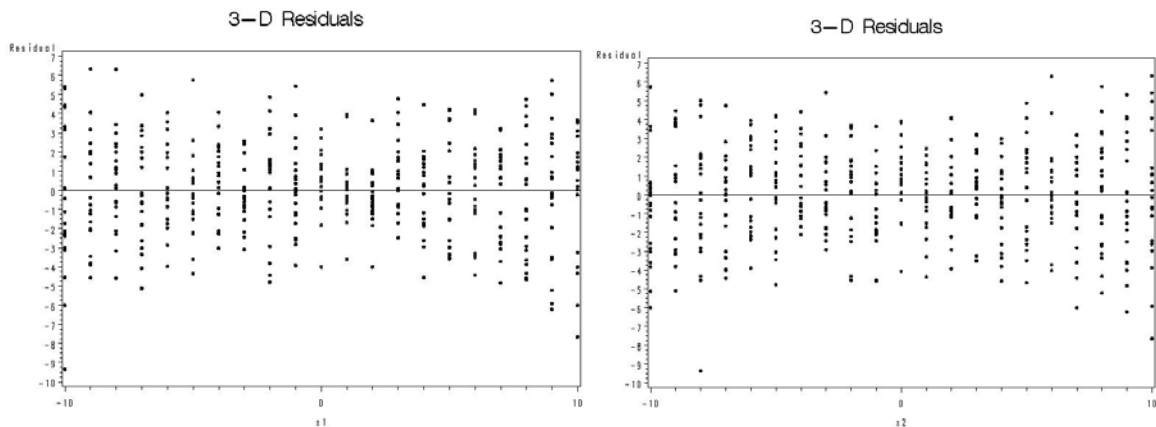
The resulting graph seems to indicate that changes in both x1 and x2 affect the response Y.

Figure 19. Y versus (x1,x2)



Next a regression is run, the residuals computed and plotted against each independent variable, a practice often advocated in statistics texts. A horizontal line at 0 separates the positive from negative residuals. The code follows the graph.

Figure 20. Residuals Versus Independent Variables



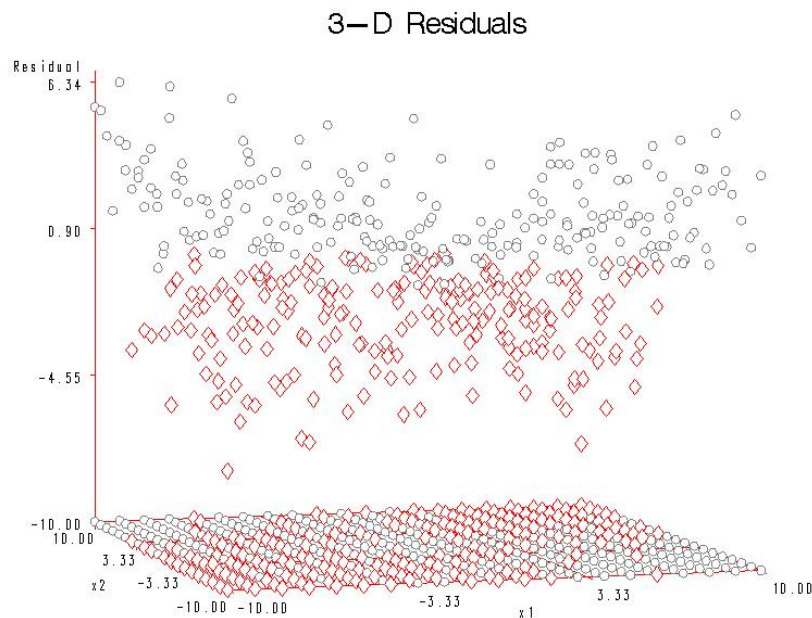
```
proc reg data=a;
  model Y = x1 x2;
  output out=out1 residual=r;
run;
proc gplot; plot r*x1 r*x2/vref=0;
  symbol1 v=dot i=none;
  Title "3-D Residuals";
```

```
run;
```

There is a slight increase in spread on the left and right edges of both plots but nothing really extreme. To investigate further, the next program colors the positive and negative residuals different colors as well as using two different shapes and sizes. The resulting 3 dimensional plot in figure 21 appears to have the larger and darker (red) diamonds concentrated in two opposing corners and the smaller, lighter (gray) circles in the other two. This is an indication of interaction, that is, a product term $x_1 \times x_2$ should be added to the model to improve the fit. Of course a simple plot of x_1 by x_2 using symbol statements to distinguish positive from negative residuals would contain much of the same information. Note that PROC G3D will not plot two points that are exactly above each other so in the program a slight, visually imperceptible "jittering" of the x_1 coordinate allows the floor points to appear.

```
data out1; set out1;
  color="gray "; shape="balloon"; size=0.5;
  if r<0 then do; color="red"; shape="diamond"; size=0.8; end; output;
  r=-10; x1=x1+.00001*(ranuni(12345)-.5); output;
proc g3d; scatter x2*x1=r /
  shape = shape size=0.5 color=color tilt=85 noneedle rotate=15 size=size;
run;
```

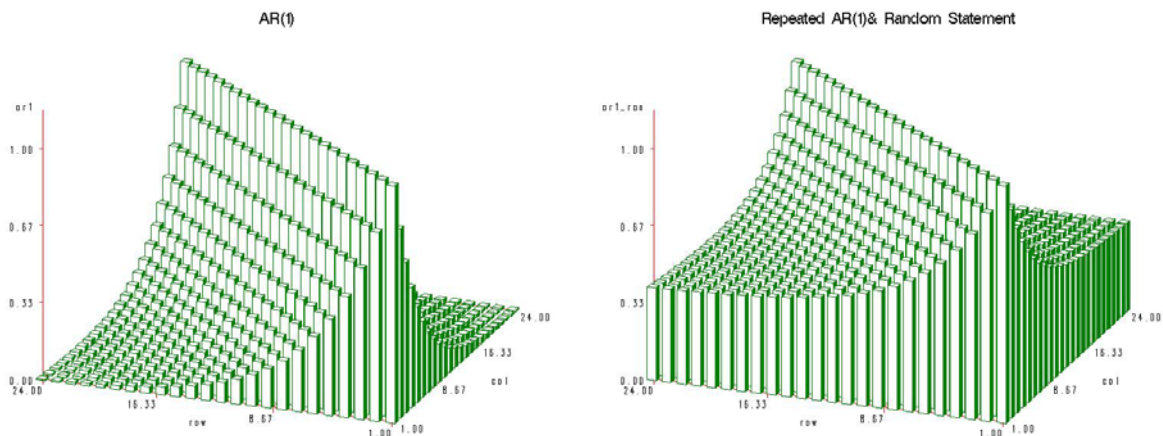
Figure 21. Residuals from Surface Fit.



MIXED MODEL COVARIANCE STRUCTURES Mixed models (PROC MIXED) can contain various variance structures specified in either a RANDOM or a REPEATED statement. Keeping track of what the various matrices look like can be a challenge. Also, identifiability problems can arise when both the RANDOM and REPEATED statements are used. Typically this would occur when one has repeated measures on subjects and a random subject effect. The random subject effect essentially raises all the elements of the type of covariance structure specified in the REPEATED statement. If the resulting modified matrix is in the same class as the type specified in the REPEATED statement, convergence problems will likely ensue. These issues can be illustrated nicely in a 3D graph. The position in the floor represents the row and column number of the covariance matrix and the height represents the correlation, typically between two times given by the row and column numbers. The following figures show several common correlation matrices used in PROC MIXED. Some of these include effects of a RANDOM statement along with the REPEATED statement. In the first figure, we see an AR(1) structure and next to it, an AR(1) plus random subject effect. The left graph declines exponentially to 0 as the row and column numbers get farther apart, the defining feature of AR(1). In the right graph there is exponential decay but not to 0. Thus the AR(1) type in REPEATED statement is compatible with a random subject effect whereas an unstructured matrix in the REPEATED statement, for example, would

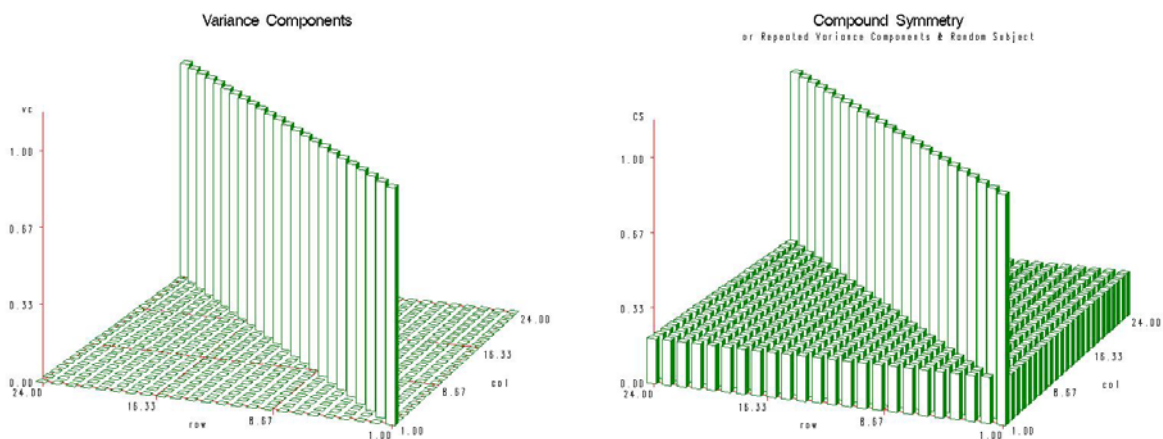
remain unstructured if all elements were increased by the same amount.

Figure 22. Repeated / type=AR(1) (without and with a random subject effect) .



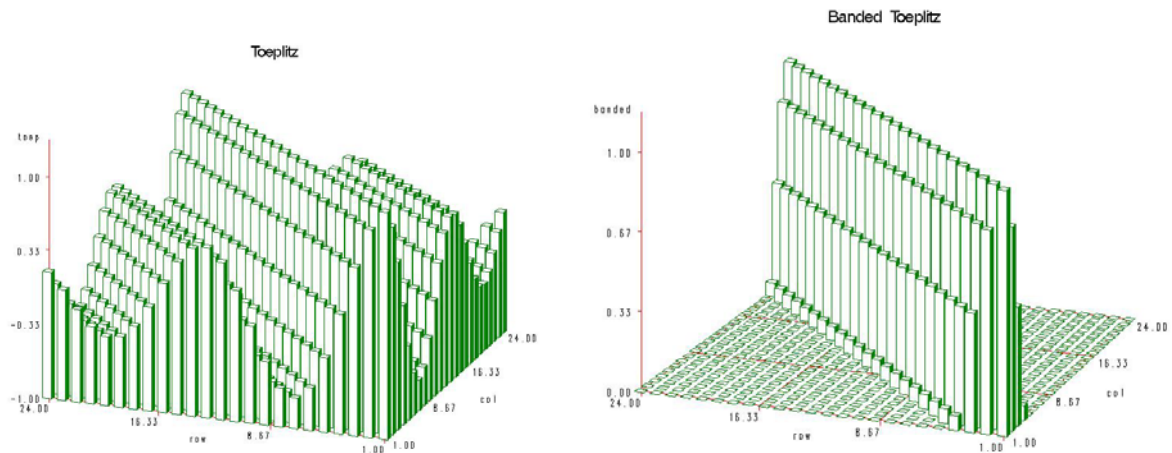
The variance component structure is a common one in the REPEATED statement as is the compound symmetry structure. A random subject effect added to the first gives the second of these two types. It also becomes clear that compound symmetry in the REPEATED statement is incompatible with a random subject effect (it would be overparameterized).

Figure 23. Variance Component and Compound Symmetry.



Toeplitz and banded Toeplitz structures have a common value a on the diagonal, another value b just above and below that, another value c just above and below that, etc. A banded Toeplitz has bands of 0 after some point. Again, a picture can help illustrate.

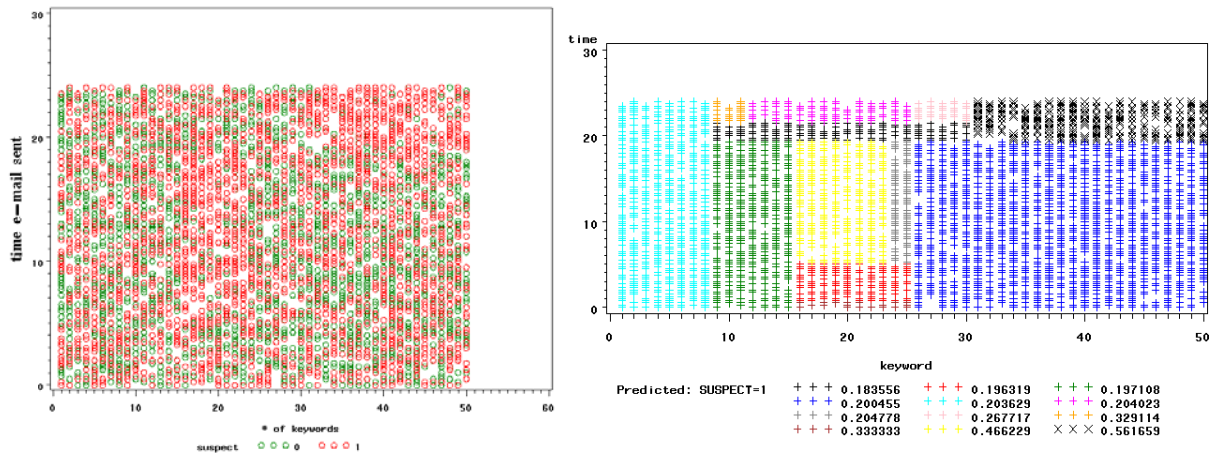
Figure 24. Toeplitz and Banded Toeplitz.



All PROC MIXED variance structure graphs are straightforward PROC G3D runs using the SCATTER statement and the option /SHAPE = “PRISM”;

DATA MINING A main tool in data mining is recursive splitting or so-called “trees”. In the last illustration, some artificial data on e-mail messages was generated. This was presented at the SAMSI Undergraduate Workshop to show undergraduates from around the country with mathematical interests how the basic tree idea works. The scenario is that e-mail messages are meticulously analyzed by experts and classified as threats or nonthreats based on their content. At the same time a list of keywords was developed and the time of day each message was sent was recorded. The idea is to use these two “features” of the data to pre-screen messages in the future so that the more likely threats would be investigated first. The idea of what a tree produces is nicely illustrated by two plots. First, a plot with suspect messages indicated by red and nonsuspect messages by green circles is shown on the left in figure 25. There seem to be some “hot” areas but they are not clearly delineated. The recursive splitting tree produces vertical and horizontal line segments to divide the plot into rectangles that are relatively homogeneous with respect to the proportion of suspect messages. The graph on the right shows the results. These are both simple PROC GPLOT graphs with symbol statements given the color.

Figure 25. A Data Mining Example.



CONCLUSION

This paper illustrates that graphs are not only useful for data presentation but also for presenting statistical concepts to clients. Graphics can reveal features of data that are not readily apparent in calculations. This paper has shown some examples of the annotate facility, templates, 3D graphs, maps with annotation, and some straightforward gplot graphs to help analyze regression models.

ACKNOWLEDGMENTS

I am indebted to the SAMSI institute Anomaly Detection Working Group for the data mining example. Work on the fish recruitment data was supported by the CWISE project at NCSU.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

David A. Dickey
North Carolina State University, Dept. of Statistics
Box 8203
Raleigh, N. C. 27609-8203
(919)-51501925
Email: dickey@stat.ncsu.edu
Web: <http://www4.stat.ncsu.edu/~dickey/>

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

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