User's Manual for the Gaussian Windows Program

RIACS Technical Report 92.05
February 1992

Louis A. Jaeckel

Work reported herein was supported in part by Cooperative Agreements NCC 2-408 and NCC 2-387, between the National Aeronautics and Space Administration (NASA) and the Universities Space Research Association (USRA)>

RIACS

Technical Report 92.05

February 1992

USER'S MANUAL FOR THE GAUSSIAN WINDOWS PROGRAM

by

Louis A. Jaeckel

ABSTRACT

"Gaussian Windows" is a method for exploring a set of multivariate data, in order to estimate the shape of the underlying density function. The method can be used to find and describe structural features in the data. The method is described in two earlier papers. I assume that the reader has access to both of those papers, so I will not repeat material from them. The program described herein is written in BASIC and it runs on an IBM PC or PS/2 with the DOS 3.3 operating system. Although the program is slow and has limited memory space, it is adequate for experimenting with the method. Since it is written in BASIC, it is relatively easy to modify. The program and some related files are available on a 3-inch diskette. A listing of the program is also available. This user's manual explains the use of the program. First, it gives a brief tutorial, illustrating some of the program's features with a set of artificial data. Then, it describes the results displayed after the program does a Gaussian window, and it explains each of the items on the various menus.

USER'S MANUAL FOR THE GAUSSIAN WINDOWS PROGRAM

1. INTRODUCTION

Gaussian windows is a method for exploring a set of multivariate data, in order to estimate the shape of the underlying density function. The method can be used to find and describe structural features in the data. The method is described in Jaeckel (1990) and Jaeckel (1991b). I assume that the reader has access to both of these papers, so I will not repeat material from them. (A brief review of the method is also given in Jaeckel, 1991a.)

The program described herein is written in BASIC and it runs on an IBM PC or PS/2 with the DOS 3.3 operating system. Although the program is slow and has limited memory space, it is adequate for experimenting with the method. Since it is written in BASIC, it is relatively easy to modify. The program is available on a 3" diskette. Also on the diskette are some data files and a related file. A listing of the program is also available.

This user's manual explains the use of the program. Section 2 gives a brief tutorial, illustrating some of the program's features with a set of artificial data. Section 3 describes the results displayed after the program does a Gaussian window. Section 4 explains each of the items on the various menus.

2. A BRIEF TUTORIAL

In this section we will run the Gaussian windows program on a small set of artificial data. Before you begin, you should be familiar with the material in Sections 1 through 4 of Jaeckel (1991b).

SETTING UP

You will need an IBM PC or compatible with the DOS 3.3 operating system. The program should also work with newer versions of DOS. You also need a diskette containing the program and the related files. The file name of the program is GGWW.BAS. If you have a fixed disk, I suggest copying all of the files on the diskette to a subdirectory on the fixed disk. The program and the other files should all be in the same subdirectory. If the program is not compatible with your system in some way, something will probably go wrong during the tutorial, and you will have to modify the program (or your system). There may be some non-standard BASIC statements in the program.

If you want to run the program from the diskette, assumed to be in Drive A, you should have the prompt A> on the screen. If you want to run it from the fixed disk, you should be in the subdirectory containing the files from the diskette, and you should have the prompt C> on the screen. To enter the BASIC interpreter,

→ Type BASICA and press [Enter]

(To exit from BASIC, you would type SYSTEM [Enter])

In this tutorial, all of the things you should type will be preceded by → .

To load the program,

→ Press the function key [F3] and type GGWW and press [Enter]
(Or type LOAD"GGWW" [Enter])

To run the program,

→ Press the function key [F2]
(Or type RUN [Enter]) The Main Menu should appear on the screen.
(To leave the program, you would type Q [Enter] from the Main Menu.)

When choosing an option on this or any other menu, always type lower-case letters and then press [Enter] . ("Caps Lock" should be off.)

USING THE PROGRAM

The Main Menu should be on the screen. To load a data file into memory,

→ Type (lower-case) L [Enter]
A list of files should appear on the screen. The file named TUTR contains a small set of artificial data.

→ Type TUTR [Enter]
The screen will show that P = 3 and N = 125. The Main Menu should reappear.

→ Type N [Enter]
The program will compute the sample mean, sample standard deviation, minimum, and maximum for each variable. It will then ask you if you want to normalize the data. Don't do it:

→ Type N [Enter]
The Main Menu should reappear.

To explore the data with Gaussian windows,

→ Type W [Enter]

The Window Menu will appear. We will begin with an "infinite" window; that is, all data points will have equal weight. See Jaeckel (1991b), p. 17. In this case the program will do a standard principal components analysis. With an "infinite" window, the window center doesn't matter, so

→ Type S [Enter]

This will keep the window center shown. The program will now ask you for the WSD, the window standard deviation. Enter a zero, for an "infinite" WSD:

→ Type 0 [Enter]

The program will now compute $\bar{x}_w$, which in this case is the unweighted sample mean vector, and S_w, the sample covariance matrix. After every 50 data points it displays a * . It then inverts S_w and finds the eigenvalues and eigenvectors of S_w. The eigenvalues are then converted to the eigenvalues of the matrix $\hat{B}$. See Jaeckel (1991b), p. 18.

Look at the results. They are described in Section 3 below. To continue,

→ Press [Enter]

The Results Menu should appear on the screen.

We will now move the window center to the unweighted sample mean, by moving along each eigenvector, or principal axis.

→ Type A [Enter]

The "mean" shown is the distance from the window center to the projection of the sample mean onto the first eigenvector.

→ Type M [Enter]

This moves the window center that distance along the first

eigenvector. Repeat this for each of the eigenvectors:

→ Type M [Enter]

→ Type M [Enter]

The Results Menu should reappear. The window center should now be at the sample mean vector, which is approximately (.53, 1.19, .76). (If it isn't, you can restore the original window center by typing the letter O [Enter] from the Results Menu. Then try again.)

→ Type W [Enter]

The Window Menu will reappear, with the new window center displayed. (Optional: If you want to see $\bar{x}_w$ and S_w when they are computed, type T [Enter] . To reverse this option, type T [Enter] when you return to the Window Menu.)

→ Type S [Enter]

This keeps the window center. For the WSD, enter zero again, for an "infinite" window:

→ Type 0 [Enter]

Lood at the results. Some of the results will be the same as before, and some will be different. The "means" and "derivatives" should now all be very near 0. To continue to the Results Menu,

→ Press [Enter]

With the Results Menu on the screen,

→ Type R [Enter]

This will display the results again.

→ Press [Enter] to return to the Results Menu.

Now we will do a cross-tab. See Jaeckel (1991b), p. 19. The program will project the data points onto the plane generated by the first two principal axes. We will choose a rectangle in the plane.

The rectangle will be divided into a 16-by-16 array of "bins". The rest of the plane will be divided into bins which are semi-infinite rectangles. The number of data points falling in each bin will be displayed. With the Results Menu on the screen,

→ Type C [Enter]

We now define the rectangle by entering a minimum and a maximum for each of the principal axes. For the first principal axis, enter two numbers with a comma between them:

→ Type -4 , 4 [Enter]

The first principal axis will be the horizontal axis in the cross-tab. For the second axis,

→ Type -4 , 4 [Enter]

(You can try other numbers if you want. You don't have to use the same numbers on both axes.) The program then asks you for a 1 or a zero. For the "no box" option,

→ Type 0 [Enter]

The "box" option will be explained in Section 4. After a pause to run through the data set, the program will display the cross-tab.

Look at the cross-tab. There appear to be two clusters in the data. Since each side of the rectangle we chose is 8 units long, each bin is a .5 by .5 square. By counting from the center of the picture, we see that the center of one of the clusters is roughly a distance of 2.5 to the right of the center of the cross-tab, which is at the window center. The center of the other cluster is roughly 1.5 to the left of the center. So both of the cluster centers appear to be near the first principal axis. Note that most of the points in each cluster appear to be within a distance of about 2 from the

cluster center.

→ Press [Enter] to return to the Results Menu.

THE CLUSTER ON THE RIGHT

We will search for the cluster on the right first. We will move the window center from its present location (at the unweighted sample mean) a distance of 2.5 along the first eigenvector.

→ Type A [Enter]

To move along the first eigenvector,

→ Type 2.5 [Enter]

Since we do not want to move along the other eigenvectors, enter a zero for the second eigenvector:

→ Type 0 [Enter]

Do the same for the third:

→ Type 0 [Enter]

The Results Menu should reappear, and the new window center should be approximately (-.08, 3.30, 1.94).

To go to the Window Menu,

→ Type W [Enter]

To keep the window center,

→ Type S [Enter]

Now we need to choose a window size. The program does only spherical Gaussian windows, as explained in Jaeckel (1991b), pp. 16-17. The WSD is a parameter for the size of the window. Since most of the cluster is within about 2 of the cluster center, I would begin with a WSD about half of that distance, or a little larger. For the WSD,

→ Type 1.2 [Enter]

The program will now do a true Gaussian window.

Look at the results. These results indicate that there appears to be a local maximum, or cluster center, in the window region. This is because none of the "means" is much larger than the WSD, and all of the "SDs" (for "standard deviations") along the eigenvectors are positive and not much larger than about twice the WSD. Each "mean" is the distance along an eigenvector from the window center to the projection of the apparent peak onto that eigenvector. See Jaeckel (1990), p. 39, and Jaeckel (1991b), p. 14.

Since the "means" are not very large, we will move the window center to the apparent peak. The Results Menu should be on the screen. You may have to press [Enter] to get there. To alter the window center,

→ Type A [Enter]

To move along the first eigenvector a distance equal to the first "mean",

→ Type M [Enter]

(This is easier than typing in the number.) Do the same for the other two eigenvectors:

→ Type M [Enter] M [Enter]

The Results Menu should reappear. The new window center should be at the apparent peak, which is approximately (-.02, 3.02, 1.89).

To return to the Window Menu,

→ Type W [Enter]

→ Type S [Enter] to keep the window center.

Now choose a WSD. Since the "SDs" we found in the previous window were not large, I would try WSD = 1.1:

→ Type 1.1 [Enter]

(You can try other values if you want.) The program will now do the computations for this window.

Look at the results. They should indicate a peak in the window region. The "means" should now be fairly small. Move the window center to the new estimated peak:

→ Type A [Enter] M [Enter] M [Enter] M [Enter]

The Results Menu should reappear.

Go to the Window Menu as we did before, keep the new window center, and try a smaller window. Since the "SDs" were not large, use .9 for the WSD. Look at the results. The "means" should be small, indicating that the window center is close to the center of the cluster.

Repeat this process once or twice more: Alter the window center to the new estimated peak, go to the Window Menu, and use .9 for the WSD. The window centers should converge to a point which is approximately (-.09, 3.12, 1.98). When the window center is at the peak, the "means" and the "derivatives" should all be very near 0. The estimated density at that point is .54, and the estimated *cluster mass* is .41.

This window center is a *center point* for the data set. See Jaeckel (1991b), pp. 20-21. We will save the information describing this center point. I assume that you now have the Results Menu on the screen, after running a window whose center is at the peak and whose WSD is .9.

→ Type S [Enter]

Since this center point is a local maximum, enter a zero when you are asked to:

→ Type 0 [Enter]

The point is now saved as Center Point #1. (That is, it is saved in memory, not on the disk.) The Results Menu should reappear.

Do a cross-tab centered on this cluster. From the Results Menu,

→ Type C [Enter]

Choose a minimum and a maximum for each of the two principal axes. I usually use values that are 2 or 3 times the corresponding "SDs".

→ Enter "Min" [comma] "Max" [Enter]

Enter a zero for the "no box" option:

→ Type 0 [Enter]

The cross-tab should show a cluster, and also a large number of data points (the rest of the data) off to one side. Press [Enter] to return to the Results Menu.

Note that these numbers are "soft" in the sense that they depend on the WSD. If we had used a somewhat different WSD, we would have gotten similar, but different, results. (Optional: Try a window centered at this peak, but with a different WSD.)

THE CLUSTER ON THE LEFT

Now we will look for the center of the other apparent cluster in the first cross-tab that we did. Recall that the center of that cluster appeared to be roughly 1.5 to the left of the center of the cross-tab. We will have to go back to the first set of principal axes that we found. Return to the Window Menu:

→ Type W [Enter]

To enter a new window center,

→ Type N [Enter]

Enter the unweighted sample mean (.53, 1.19, .76) by typing each

number at the "Coord" prompt:

→ Type .53 [Enter] 1.19 [Enter] .76 [Enter]

→ Type 0 [Enter] for the WSD.

Continue to the Results Menu. To move the window center 1.5 to the left (in the negative direction) along the first eigenvector,

→ Type A [Enter]

→ Type -1.5 [Enter]

Since we do not want to move along the other eigenvectors,

→ Type 0 [Enter] 0 [Enter]

The new window center should be approximately (.89, -.08, .05).

Return to the Window Menu. Keep this window center, and try a window with a WSD of 1.2. Look at the results for this window. There should be an apparent peak in the window region (all "SDs" positive and not too large, and all "means" fairly small.) From the Results Menu, move the window center to the apparent local maximum:

→ Type A [Enter] M [Enter] M [Enter] M [Enter]

The Results Menu should reappear.

Go to the Window Menu and run a window with this center and a WSD of 1. Repeat this process a few times; that is, move the window center to the new peak, return to the Window Menu, and do a window with WSD = 1. The window centers should converge in a few steps, to approximately (.89, .14, -.06). When the window center is at the peak, all of the "means" and "derivatives" will be nearly 0.

This window center is a center point for the data set. From the Results Menu, save the information describing it:

→ Type S [Enter]

As before, enter a zero to indicate that this point is a local

maximum:

→ Type 0 [Enter]

The point is now saved as Center Point #2. (Optional: From the Results Menu, do a cross-tab of this cluster.)

THE SADDLE POINT

The data set seems to consist of two slightly overlapping clusters. In the region where they overlap, the density function should appear to have a saddle point. See Jaeckel (1991b), pp. 42-44. We will now look for that point and add it to our list of center points. A simple way to choose a starting point for our search is to take the midpoint of the line segment joining the two peaks above. That point is approximately (.40, 1.63, .96). (Another way to choose a starting point would be by looking at a cross-tab.)

Go to the Window Menu. To enter the coordinates of this point,

→ Type N [Enter]

→ Enter each coordinate, one at a time, at the prompt.

Since we may not be very close to the saddle point, I would use a relatively large WSD here, say 1.5:

→ Type 1.5 [Enter]

Look at the results. The first "SD" is negative, indicating that the density function is concave upward along the first eigenvector. The other "SDs" are positive. This suggests that we are looking at a *bar*, which is an extended structure running through the window region. What we have here is a shape like a bar joining the two clusters; the saddle point will be a point of minimum density along the center line of this bar. See Jaeckel (1991b), pp. 42-44.

Note that the estimated cluster mass is undefined in this case; this is indicated by a -1 on the screen.

We will now move the window center to the nearest point on the apparent center line of the bar; that is, we will move in a direction orthogonal to the center line. From the Results Menu,

→ Type A [Enter]

Since the first eigenvector is parallel to the estimated center line, we will not move along it:

→ Type 0 [Enter]

We will, however, move along each of the other eigenvectors, which are orthogonal to the estimated center line. For each, we will move to the point of maximum estimated density along that direction; that is, we will move a distance equal to the "mean" for that eigenvector:

→ Type M [Enter] M [Enter]

The Results Menu should reappear. The new window center should be approximately (.54, 1.72, .89).

Go to the Window Menu and run another window. Keep this new window center and reduce the WSD to 1.2. Look at the results. The first "SD" is negative and the others are positive. The second and third "means" are small, indicating that the window center is near the estimated center line of the bar. The second and third "SDs" give us an idea of the shape of the cross section of the bar.

Since the first "mean" is also relatively small, compared to the WSD, there appears to be a saddle point in the window region, and we can move the window center to that point. (If the first "mean" were larger, we would try to find the saddle point by moving along the center line in a series of steps, in the direction of decreasing

density.) Since the first "SD" is negative, the first "mean" is the distance along the first eigenvector to the point of *minimum* estimated density in that direction, rather than to the maximum. See Jaeckel (1991b), p. 14. So, from the Results Menu, we can move the window center to the estimated saddle point as follows:

→ Type A [Enter] M [Enter] M [Enter] M [Enter]

(This is just like moving to a local maximum, which we did earlier.) The new window center should be approximately (.50, 2.07, 1.09).

Go to the Window Menu, keep this center, and reduce the WSD to 1. Look at the results. Since the first "SD" is negative, and since all of the "means" are small, the saddle point appears to be in the window region. Move the window center to the estimated saddle point:

→ Type A [Enter] M [Enter] M [Enter] M [Enter]

The new window center is approximately (.60, 1.96, .96). Note that since the second and third "SDs" are small (and positive), we can reduce the window size and still capture the cross section of the bar in a window.

Go to the Window Menu, keep the new center, and reduce the WSD to .8. Look at the results. Move the window center again to the estimated saddle point, as we did above, go to the Window Menu, and run another window with WSD = .8. All of the "means" and the "derivatives" should be near 0, indicating that the window center is now very close to the saddle point. Repeat this process once or twice more; that is, move the window center as before and run another window with WSD = .8. The window center should now be at the saddle point. Its location is approximately (.69, 1.94, .89). The estimated density at this point is .060. (If we had used a different

WSD, the results would be somewhat different.)

We will save this saddle point as a center point for the data set. From the Results Menu,

→ Type S [Enter]

Since a saddle point is considered to be a point on the center line of a bar, enter a 1 instead of a 0 at the prompt:

→ Type 1 [Enter]

The point is now saved as Center Point #3. (Optional: Do a cross-tab of the region around the saddle point.)

CENTER POINTS AND OUTLIERS

We will now count the data points *associated* with each of the three center points, and also count the *outliers* (data points not associated with any center point). See Jaeckel (1991b), pp. 20-28. I assume that you have saved the three center points found above, and that the Results Menu is on the screen. Go to the Window Menu and then to the Main Menu:

→ Type W [Enter] M [Enter]

(Optional: If you want, you can store the information on the saved center points in a disk file. To do this, type SC [Enter] from the Main Menu. Make up a new file name and enter it. This information is already stored in the file named TUTRCP. Your center points may differ slightly from mine, but that doesn't matter.)

The Main Menu should be on the screen. To compute the *M-distance* (Jaeckel, 1991b, pp. 22-26) from each data point to each center point, go to the Center Point Menu:

→ Type CP [Enter]

The Center Point Menu will appear. It should say that you have saved

three center points. To run through the data set,

→ Type D [Enter]

Wait while the program runs through the data set. (For a large data set this is a time-consuming operation.) For each data point, the program tests whether it is associated with each of the center points. At the end, you will see that there are four outliers, 49 data points associated with the first cluster, 73 associated with the second cluster, and eight associated with the saddle point. (If your center points are different from mine, you might get slightly different results here.)

The Center Point Menu should reappear. Note the information displayed for each data point. (Optional: You can find this information for an individual data point by typing X [Enter] and following the instructions.) To find which data points are the outliers, enter the letter O:

→ Type O [Enter]

The program will run through the data set and stop at each outlier. To continue after each outlier,

→ Press [Enter]

The Center Point Menu will eventually reappear.

To return to the Main Menu,

→ Type M [Enter]

This is the end of the tutorial. You are on your own now.

To exit from the program,

→ Type Q [Enter]

To exit from the BASIC interpreter,

→ Type SYSTEM [Enter]

3. RESULTS DISPLAYED AFTER RUNNING A WINDOW

After doing the computations for a Gaussian window, the program displays the results described below. Since the results will not fit on one screen, you will have to press [Enter] once or twice to continue to the Results Menu.

As the program goes through the data it displays a * after every 50 data points. Those data points that are very far from the window center are excluded: If the expression $Q = (x - a)'V(x - a)$ in the exponent of $w(x)$ (Jaeckel, 1991b, p. 7) is greater than 30, the data point is skipped. The program displays the number of skipped data points after "N with Q > 30".

If you chose to have the weighted sample mean and covariance matrix displayed (from the Window Menu), they are displayed next.

Then the program inverts the weighted covariance matrix S_w and finds the eigenvectors and eigenvalues of S_w. It displays the eigenvalues as it finds them.

On rare occasions the program may fail to find all P eigenvalues of the matrix. In that case an error message will appear on the screen, and the program will skip some of the computations. Since the eigenvalue algorithm used in the program is a fairly primitive one (you may want to replace it with a better one), it might fail for a variety of reasons. Also, an "Overflow" message may occasionally appear on the screen. If either of these things happens, the best thing to do is to make a small change in the window parameters and to try another window. That should usually take care of the problem.

However, there is one situation where this will not work. If the data points lie in a linear manifold of dimension lower than P, then the covariance matrix will be singular, and the algorithm will not be able to find all of the eigenvalues. Also, some of the other computations cannot be done in this case. The program is not designed to handle such data sets. This issue is raised in Jaeckel (1990), p. 25, where I simply excluded data sets of this kind. The thing to do in such a case is to define a new coordinate system of lower dimension for the data set, so that the data points do not lie in a linear manifold.

The program then converts the eigenvalues of S_w to the eigenvalues λ_j of the matrix $\hat{B}$. See Jaeckel (1991b), p. 18. On rare occasions a cryptic message might appear at this point, saying that something has been "reset" to a new value. This is to avoid some possible numerical problems. See Jaeckel (1991b), pp. 18-19. These messages can generally be ignored.

The program then displays the following:

- The number N of data points.
- The window standard deviation WSD.
- The sum of the weights w_i attached to the data points.
- The estimated density $\hat{h}$ at the window center. See Jaeckel (1990), p. 36.
- The estimated *cluster mass* $\hat{c}$. See Jaeckel (1990), p. 29, and Jaeckel (1991b), p. 10. The cluster mass is computed only if all of the λ_j are positive, that is, if there appears to be a cluster in the window region. Otherwise it is set to -1.
- The number of eigenvectors found (normally P).

Then the program displays a table of results. Each row of the table corresponds to an eigenvector. In the "eigenvalue" column is λ_j, the eigenvalue of $\hat{B}$. The order of the eigenvalues is from smallest to largest. Thus the first row is for the eigenvector, or principal axis, along which the data points are most spread out, and the last row is for the eigenvector along which the data points are least spread out.

The "derivative" is t_j, the first partial derivative at the window center of $\log \hat{f}(x)$ along the eigenvector. See Jaeckel (1990), p. 41.

The "mean" is the distance $\frac{t_j}{\lambda_j}$ from the window center to the maximum or minimum of the density function along the eigenvector. See Jaeckel (1990), pp. 38-39, and Jaeckel (1991b), p. 14.

The "SD", if $\lambda_j > 0$, is the standard deviation $\lambda^{-1/2}$ of the univariate Gaussian function which is the component of the estimated density function along the eigenvector. If $\lambda_j < 0$, then the estimated density function in the window region is concave upward along the corresponding eigenvector, and the "SD" is a scale parameter analogous to the standard deviation for the "concave Gaussian" function along the eigenvector. See Jaeckel (1990), p. 38, and Jaeckel (1991b), p. 14. In the latter case the "SD" is displayed as a negative number; that is, the "SD" displayed is $-(-\lambda_j)^{-1/2}$. (If all of the "SDs" are positive, the largest "SD" is listed first; if some "SDs" are negative, the negative "SD" with smallest absolute value is listed first.)

The "SU" is the distance along the eigenvector from the window

center to the maximum or minimum expressed in "standard units", that is, the "mean" divided by the absolute value of the "SD".

The program then asks you to press [Enter] . Then it displays the eigenvectors, one to a line on the screen, in the order of the corresponding eigenvalues in the table above.

Finally, the Results Menu appears on the screen.

4. THE MENUS

Items on all menus are chosen by typing lower-case letters followed by [Enter] . In this section, menu items are underlined.

THE MAIN MENU

→ L [Enter]

To load a data file from the disk. The program will display a list of the files in the current subdirectory.

→ Enter a file name and press [Enter]

(Or, to go back to the Main Menu, just press [Enter]) The program will load the data in the file into memory, wiping out any data that may have been there, and also any saved center points. The program expects a BASIC sequential file, in the format described below. The same format is used when the program writes files, using the S option on the Main Menu. Since such a file is actually a standard DOS file, a file with this format can be created by other means. If you have a data set you want to explore, you will have to convert it to the expected format (or modify the program).

The order in which the numbers are written on a data file is as

follows: First is P, the dimension of the space (the number of variables). Next is N, the number of data points. Then, the first coordinate of the first data point, followed by the second coordinate of the first data point, and so on, through the P^{th} coordinate of the first data point. Then comes the first coordinate of the second data point, and so on. Each number is formatted as a floating point decimal number and is encoded in ASCII, just as it would be displayed on the screen by a BASIC program. Following each number are the two bytes 0D(hex) 0A(hex) — that is, "carriage return" and "linefeed". At the end of the file there is the byte 1A(hex), the DOS end-of-file marker. With this information, you can convert a data file to the format expected by the program. Note that because of the limited memory space, you must have $P \leq 7$ and $N \leq 650$.

→ W [Enter]

To go to the Window Menu, from which you can run a Gaussian window on the data currently in memory.

→ G [Enter]

To go to the Data Generation Menu, where you can generate a set of artificial data consisting of random points from a mixture of multivariate Gaussian clusters. This option wipes out whatever data may be in memory, and also any saved center points.

→ A [Enter]

To add more artificial data to the data already in memory. This option also goes to the Data Generation Menu.

→ S [Enter]

To save the data currently in memory to a disk file. The program

will display a list of the files in the current subdirectory.

→ Enter a file name and press [Enter]

(Or, to go back to the Main Menu, just press [Enter]) If you enter the name of an existing file, whatever is now in it will be lost. The program will ask you how many variables to save. This number would usually be P, but it could be more or less, depending on what is in memory.

→ Enter "number" [Enter]

(You can escape here by entering 0.) If K is the number you enter, the program will write the first K variables to the disk file. The format of the file will be as described above, under the L option.

A case where you might want to save more than P variables is where $P < 7$ and you have used the program to determine which data points are outliers. Since Variable 7 is used to indicate which data points are outliers (see below), you can save the results by saving seven variables in a data file.

→ N [Enter]

To compute, for each variable separately, the sample mean, standard deviation, minimum, and maximum. The program will then ask you if you want to normalize the data, that is, for each variable, to subtract the mean and divide by the standard deviation for that variable. Enter Y if you do, or N if you don't:

→ Type Y [Enter] or Type N [Enter]

→ C [Enter]

To do a cross-tab of the data, that is, to define two new variables and to map the data onto the plane defined by those two variables.

The plane is divided into an array of "bins", or cells, and the number of data points falling in each bin is counted. Each of the two new variables can be one of the existing variables (including Variable 7 when used to indicate outliers), or it can be a linear combination of the existing variables. The program will ask you to enter a number for the X variable, which will be the horizontal axis of the cross-tab. If you want to use one of the existing variables, enter a number between 1 and 7; if you want to create a linear combination of the existing variables, enter a zero:

→ Enter "number" [Enter]

If you entered a zero, the program will ask you for P coeffients to define the linear combination. Enter each coefficient at the prompt:

→ Enter "number" [Enter]

To define the left and right boundaries of a rectangle in the plane, enter two numbers, with a comma between them:

→ Enter "Min" [comma] "Max" [Enter]

→ Do the same steps for the Y variable, the vertical axis of the cross-tab.

If P < 7, the program will tell you to "Enter 0 for all points, 1 for outliers only". If you want the cross-tab to include all of the data points, enter a zero; if you have already determined which data points are outliers and you want the cross-tab to include only the outliers, enter a 1:

→ Enter "number" [Enter]

The rectangle you defined above is divided into a 16-by-16 array of bins. The rest of the plane is divided into semi-infinite rectangular bins by extending the lines defining the bins in the

rectangle. The program counts the number of data points (or outliers) falling in each bin and then displays the results. If the number in a bin is greater than 99, it is displayed in reverse video (black on white).

You now have some options. To return to the Main Menu,

→ Press [Enter]

If you want to repeat the cross-tab using the same two variables but with different minima and maxima to define the rectangle, or if you want to change the "outlier" option,

→ Type R [Enter]

Then, for each variable, enter a new minimum and maximum:

→ Enter "Min" [comma] "Max" [Enter]

→ Enter 0 or 1 for the outlier option (if P < 7).

The TRACE option: If you want to do a trace — that is, if you want to find out which data points (or outliers) fall in a particular bin in the cross-tab,

→ Type T [Enter]

You must now enter the coordinates of the bin to be traced. The horizontal axis is X, going from left to right. In each row, the semi-infinite bin on the left is numbered 0, the middle 16 bins are numbered 1 through 16, and the semi-infinite bin on the right is 17. The vertical axis is Y, going from bottom to top, as on a graph of the XY-plane. In each column, the bottom bin is numbered 0, the middle 16, going upward, are 1 through 16, and the top bin is 17. Enter the X and Y coordinates of the bin you want:

→ Enter "X" [comma] "Y" [Enter]

→ Enter 0 or 1 for the outlier option (if P < 7).

The program will display the number of each data point (or outlier) falling in the bin. (If you want to see the coordinates of a particular data point, you can do that by using the X option on the Window Menu, the Results Menu, or the Center Point Menu.)

→ Press [Enter] to continue.

→ P [Enter]

To change the number of variables to use. If you give P a new value, say K, the program will use the first K variables for the data now in memory. If you don't want to change P, enter its current value. (You must enter something.)

→ Enter "number" [Enter]

An example is the *IRAS* data set, which is in the file named IRAS3. This is the data set explored in Jaeckel (1991b). This file contains six variables for each data point, but since I used only the first four in my analysis of the data, the program automatically sets P to 4 after loading the data into memory. However, all six variables are loaded into memory, and therefore are accessible to the program by changing P. (Variable 5 is the galactic latitude and Variable 6 is the galactic longitude.)

→ CP [Enter]

To go to the Center Point Menu, where you can compute M-distances to center points, test whether data points are associated with center points, find outliers, and look at the information saved for each center point. See Jaeckel (1991b), pp. 20-28.

→ SC [Enter]

To save in a disk file the center point information currently in

memory. The program will display a list of the files in the current subdirectory.

→ Enter a file name and press [Enter]

(Or, to go back to the Main Menu, just press [Enter]) The program will write the center point information to the disk file. The information saved is described below, under the S option on the Results Menu.

→ LC [Enter]

To load into memory the center point information previously stored in a disk file. The program will display a list of the files in the current subdirectory. Note that before going to this option, the variable P must be set to the value of P that was used when the center point file was created.

→ Enter a file name and press [Enter]

(Or, to go back to the Main Menu, just press [Enter]) The program will load the information into memory, wiping out any center point information that may have been there. Thus, if you want to add new center points to those already in a disk file, you must load the file into memory first, and then add new center points by using the S option on the Results Menu.

→ Q [Enter]

To exit from the program and return to the BASIC interpreter.

THE DATA GENERATION MENU

If you come to this menu by the G option, any data and center points in memory will be wiped out, and the program will first ask

you for a value of P, the number of dimensions. Enter a number between 1 and 7:

→ Enter "number" [Enter]

If you come to this menu by the A option, the value of P is already set, and you will add data to the existing data.

→ C [Enter]

To generate a cluster (or another cluster) of random data points with a multivariate Gaussian shape. The program will first ask you for the center of the cluster. Enter each coordinate of the center at the "Coord" prompt:

→ Enter "number" [Enter]

The cluster will be described by entering its principal axes (a set of orthogonal vectors) and the standard deviation along each axis. For each principal axis, the program first asks for the standard deviation along that axis:

→ Enter "number" [Enter]

Then the program asks for a non-zero vector to give the direction of the axis. The vector does not have to be of unit length, since the program will normalize it. However, the vectors must be mutually orthogonal. For each coordinate of the vector,

→ Enter "number" [Enter]

→ Repeat these steps for each axis.

Then, at the prompt, enter the number of points to generate for this cluster:

→ Enter "number" [Enter]

The program will generate a set of random points from this

multivariate Gaussian distribution.

The Data Generation Menu then reappears. You can generate more clusters by repeating the steps above.

→ M [Enter]

To return to the Main Menu.

THE WINDOW MENU

The first five options give you different ways of choosing a window center for a Gaussian window. Each of these options then goes to the place where you enter the *window standard deviation*, or WSD, which is a parameter for the size of the window. The program does only spherical Gaussian windows. See Jaeckel (1991b), pp. 16-17.

→ N [Enter]

To enter a new window center. Enter each coordinate of the desired center at the "Coord" prompt:

→ Enter "number" [Enter]

Go to "WSD" below.

→ S [Enter]

To keep the window center displayed at the top of the Window Menu. Go to "WSD" below.

→ A [Enter]

To add a vector to the window center displayed at the top of the Window Menu. Enter the amount to add to each coordinate of the window center at the prompt:

→ Enter "number" [Enter]

Go to "WSD" below.

→ X [Enter]

To use a data point as the window center. This option can also be used to look at the coordinates of any data point. At the prompt, enter the number of a data point, between 1 and N:

→ Enter "number" [Enter]

(Or, you can enter 0 to return to the Window Menu.) The program will display the coordinates of the data point and ask you if you want to use it as the window center. If you do,

→ Type Y [Enter]

Go to "WSD" below.

If you don't want to use this point as the window center, you have two options. If you want to go to the next data point,

→ Press [Enter]

The steps above will be repeated. Or, if you want to enter the number of another data point or return to the Window Menu,

→ Type N [Enter]

After doing that, you can enter the number of a data point, and the steps above will be repeated, or you can return to the Window Menu by entering a zero:

→ Enter "number" [Enter]

→ C [Enter]

To use one of the center points now saved in memory as the window center. The program will list the coordinates of each center point and the WSD that was used for that point. The program will then ask you for the number of the center point you want.

→ Enter "number" [Enter]

Go to "WSD" below. If, instead, you want to return to the Window Menu, enter a zero:

→ Type 0 [Enter]

"WSD": After you have chosen a window center by any of the methods above, the program will ask you to choose a value for the WSD, the common standard deviation along each axis of the spherical Gaussian window. This is a scale parameter corresponding to the size of the window. See Jaeckel (1991b), pp. 16-17. The WSD of the previous window is displayed. Enter a positive number for the WSD. Or, if you want an "infinite" window, that is, if you want each data point to have equal weight, enter a zero. In that case the program will do a standard principal components analysis.

→ Enter "number" [Enter]

(If you want to return to the Window Menu here, without doing a window, enter a negative number for the WSD.)

The program will now run through the data and do the window computations. The results displayed are described in Section 3.

→ R [Enter]

If, after having done a window, you want to jump back to the Results Menu. The results of the previous window should still be in memory, including the window center used for that window.

→ T [Enter]

If you want the program to display the weighted sample mean $\bar{x}_w$ and the weighted sample covariance matrix S_w after they are computed, or if you want the program to stop displaying them. This option acts

as a toggle switch, reversing its previous setting. When you start the program, these quantities will not be displayed.

→ M [Enter]

To return to the Main Menu.

THE RESULTS MENU

This menu appears after the results of running a Gaussian window have been displayed.

→ R [Enter]

To repeat the display of the results, beginning with the number N of data points.

→ A [Enter]

To alter the window center by moving it from its current location along the eigenvectors just found. The eigenvector along which the data points are most spread out is listed first. For each eigenvector, or principal axis, the program repeats the line from the table displayed earlier, containing the "mean", the "SD", etc. To move the window center along that eigenvector, enter the positive or negative distance to move (or 0 if you don't want to move in that direction):

→ Enter "number" [Enter]

If you want to move a distance equal to the "mean" shown, as will often be the case, you can:

→ Type M [Enter] instead of entering that number.

Enter something for each eigenvector. The Results Menu will then reappear.

For example, if a local maximum (or a local minimum or a saddle point) appears to be in the window region, in which case all of the "means" will be relatively small, you can move the window center to that point by entering M for each eigenvector. If a bar appears in the window region (first "SD" negative or very large, all other "SDs" positive and relatively small, and all "means" but the first relatively small), you can move the window center to the nearest point on the estimated center line of the bar by entering 0 for the first eigenvector (which is parallel to the center line) and M for each of the others. See Jaeckel (1990), p. 52. If the window center is on or very near the estimated center line of a bar ("SDs" as above and all "means" but the first very near 0), you can move the window center some distance *along* the center line by entering that amount as the positive or negative distance to move along the first eigenvector. For each of the other eigenvectors, enter 0 (or, if the "means" are small and you want to "correct" for them, enter M). See Jaeckel (1990), pp. 52-53.

→ 0 (letter 0) [Enter]

To restore the original window center if you have changed it.

→ D [Enter]

To find the estimated density at any point, and, if there is a local maximum in the window region, the M-distance of the point from the local maximum. The window center need not be at the local maximum. Enter each coordinate of the point at the "Coord" prompt:

→ Enter "number" [Enter]

The program will compute the estimated density and the M-distance.

Note: If there is not a local maximum in the window region, the M-distance displayed will be meaningless.

→ X [Enter]

To find the estimated density at a data point, and, if there is a local maximum in the window region, the M-distance of the data point from the local maximum. At the prompt, enter the number of a data point, between 1 and N:

→ Enter "number" [Enter]

(Or, to return to the Results Menu, enter 0.) The program will display the coordinates of the data point and then do the computations as in the option above. The program will then ask you to enter the number of another data point (or 0, to return to the Results Menu):

→ Enter "number" [Enter]

→ C [Enter]

To do a cross-tab of the "scores" of the data points on the first two principal axes found by the Gaussian window. A "score" is the distance from the window center to the projection of a data point onto a principal axis. The program projects the data points onto the plane generated by the first two eigenvectors, divides the plane into bins as in the C option on the Main Menu, and counts the number of data points falling in each bin.

The program first asks you to define a rectangle in the plane, by entering a minimum and a maximum score for the first principal axis, which will be the horizontal axis in the cross-tab. At the prompt,

→ Enter "Min" [comma] "Max" [Enter]

→ Do the same for the second principal axis.

These numbers define a rectangle in the plane, which is divided into a 16-by-16 array of bins. The rest of the plane is divided into semi-infinite rectangular bins by extending the lines defining the bins in the rectangle.

Next, the program asks whether you want to include only those data points lying in a "box". If you want the cross-tab to include all of the data points, enter a zero for "no box":

→ Type 0 [Enter]

If, instead, you want the program to define a box in the P-dimensional space, as described below, and to include in the cross-tab only those data points lying in the box,

→ Type 1 [Enter]

If you choose this option, the program will define a box as follows: The box is centered at the window center, it contains the plane through the window center that is generated by the first two eigenvectors, and it includes a region in the P-dimensional space that surrounds the plane in the other $P - 2$ dimensions. For each data point, the program computes its "score" on each of the other $P - 2$ principal axes. If any of these scores is greater than 2.5 times the "SD" for that eigenvector, or less than -2.5 times the "SD", then the data point is excluded from the cross-tab.

The program displays the number of data points falling in each bin. If the number in a bin is greater than 99, it will be displayed in reverse video (black on white).

To return to the Results Menu,

→ Press [Enter]

→ S [Enter]

To call the current window center a *center point*, and to save in memory certain information about the point. See Jaeckel (1991b), pp. 20-24. Up to 25 center points can be saved in memory. Note that the center point information will be lost if you leave the program without saving it in a disk file. There are two kinds of center points. For the first kind, the window center must be a local maximum (the center of a cluster), with all "means" nearly 0 and all "SDs" positive and not too large. To save the window center as a local maximum, enter a zero at the prompt:

→ Type 0 [Enter]

For the second kind of center point, the window center must be a point on the estimated center line of a bar, or a saddle point, which is considered to be on the center line of a bar. All of the "means", expect possibly the first, must be nearly 0; the first "SD" must be negative, or large and positive; and all of the other "SDs" must be positive and not too large. For a saddle point, the window center must satisfy these conditions, and, in addition, the first "mean" must be nearly 0, and the first "SD" must be negative. To save the window center as a center point of this kind, enter 1 at the prompt:

→ Type 1 [Enter]

(If, instead, you decide not to save this point, enter a -1 at the prompt, and you will return to the Results Menu.)

The center point is given a number, and the following information is saved: The location of the point; the eigenvalues

λ_j; the eigenvectors; the WSD of the window used to compute this information; the cluster mass if the center point is a local maximum (for the other kind of center point, this is set to -1); and the estimated density at the point. You can look at this information later by using the P option on the Center Point Menu.

→ W [Enter]

To return to the Window Menu.

THE CENTER POINT MENU

With this menu you can compute the *M-distance* of a given point from each of the center points currently saved in memory, and other related information, as described under the X option below. See Jaeckel (1991b), pp. 20-28.

→ X [Enter]

To choose a data point for the computations. At the prompt, enter the number of a data point, between 1 and N:

→ Enter "number" [Enter]

(Or, to return to the Center Point Menu, enter 0.) The program will display the coordinates of the data point. Then, for each center point in memory, the program will display the number of the center point and the M-distance between the data point and the center point. If the center point is a local maximum, then the P-dimensional M-distance is computed; if it is a point on the center line of a bar (or a saddle point), then the (P-1)-dimensional M-distance is computed. See Jaeckel (1991b), pp. 22-26. In either case, if the data point is *associated at the 95% level* with the center point, as

defined in Jaeckel (1991b), pp. 23-24 and p. 26, then the M-distance is displayed in reverse video (black on white). Note that a data point may be associated with more than one center point. If the center point is a point on the center line of a bar (or a saddle point), the program displays the distance along the center line from the center point to x^*, the projection of the data point onto the center line. See Jaeckel (1991b), pp. 25-26. Then the program displays the Euclidean distance from the data point to the center point, and the WSD of the window that was used to compute the information on the center point. Finally, if the data point is not associated with any of the center points, the word "Outlier" appears in reverse video.

If you want to go on to the next data point,

→ Press [Enter]

If, instead, you want to return to the Center Point Menu,

→ Type **Q** [Enter]

→ **O** (letter **O**) [Enter]

To run through the data set, do the computations and display the results described above for each data point, and stop at each outlier (a data point not associated with any center point). This option allows you to look at the outliers. To continue with the next data point,

→ Press [Enter]

The program will continue to the next outlier, or to the end of the data set. If, instead, you want to return to the Center Point Menu,

→ Type **Q** [Enter]

If P < 7, then, for each data point, the memory location for the seventh coordinate of the data point is set to 0 if the point is not an outlier, or to 1 if it is an outlier. If this is done for all of the data points, either by this option or by the next option, then Variable 7 can be used as an outlier indicator in the C option on the Main Menu.

At the end of the data set, the program displays the number of outliers found, and then the number of data points associated with each of the center points. Then the Center Point Menu reappears.

→ D [Enter]

To run through the entire data set and do the computations described above for each data point. For a large data set with many center points, this operation takes some time. If P < 7, then Variable 7 is set as described in the option above.

At the end of the data set, the program displays the number of outliers found, and then the number of data points associated with each of the center points. Then the Center Point Menu reappears.

→ K [Enter]

To enter a point through the keyboard. The program will do the computations and display the results described above, treating the point as if it were a data point. Enter each coordinate of the point at the "Coord" prompt:

→ Enter "number" [Enter]

After the results are displayed,

→ Press [Enter] to return to the Center Point Menu.

→ C [Enter]

To choose one of the saved center points. The program will do the computations and display the results described above, treating the center point as if it were a data point. The program first lists all of the center points, giving the coordinates of each center point and the WSD used for the center point. It will then ask you for the number of a center point.

→ Enter "number" [Enter]

(Or, to return to the Center Point Menu, enter 0.) After the results are displayed,

→ Press [Enter] to return to the Center Point Menu.

→ P [Enter]

To display the saved information for a center point. Enter the number of a center point:

→ Enter "number" [Enter]

(Or, to return to the Center Point Menu, enter 0.) The program will display the coordinates of the center point; the WSD of the window used for the center point information; the estimated density at the point; the estimated cluster mass if the center point is a local maximum; for each principal axis, the eigenvalue λ_j, and, if $\lambda_j > 0$, the standard deviation (or 0 if not); and the eigenvectors.

The program then asks you for the number of another center point, or 0 if you want to return to the Center Point Menu.

→ Enter "number" [Enter]

→ M [Enter]

To return to the Main Menu.

REFERENCES

Jaeckel, L. A. (1990). Gaussian windows: A tool for exploring multivariate data. RIACS Technical Report 90.41

Jaeckel, L. A. (1991a). Gaussian windows: A multivariate exploratory method. In *Computing Science and Statistics, Proc. 23rd Symposium on the Interface,* pp. 58-60. Interface Foundation of North America

Jaeckel, L. A. (1991b). Using Gaussian windows to explore a multivariate data set. RIACS Technical Report 91.22