USGS-OFR-89-139 USGS-OFR-89-139

UNITED STATES
DEPARTMENT OF THE INTERIOR
GEOLOGICAL SURVEY

FREQFIT -- COMPUTER PROGRAM WHICH PERFORMS NUMERICAL REGRESSION AND STATISTICAL CHI-SQUARED GOODNESS OF FIT ANALYSIS

by

Gregg S. Hofland and Christopher C. Barton

Open File Report 89-139

Prepared in cooperation with the
Nevada Operations Office
U.S. Department of Energy
(Interagency Agreement DE-AI08-78ET44802)

Denver, Colorado
1990

Copies of this Open-File Report
may be purchased from

Books and Open-File Reports Section
Branch of Distribution
U.S. Geological Survey
Box 25425 Federal Center
Denver, Colorado 80225

PREPAYMENT IS REQUIRED

Price information will be published
in the monthly listing
"New Publications of the Geological Survey"

FOR ADDITIONAL ORDERING INFORMATION

CALL: Commercial: (303) 236-7476
FTS: 776-7476

USGS-OFR-89-139

USGS-OFR-89-139

UNITED STATES
DEPARTMENT OF THE INTERIOR
GEOLOGICAL SURVEY

FREQFIT -- COMPUTER PROGRAM WHICH PERFORMS NUMERICAL REGRESSION AND STATISTICAL CHI-SQUARED GOODNESS OF FIT ANALYSIS

by

Gregg S. Hofland and Christopher C. Barton

U.S. Geological Survey, Denver, Colorado

CONTENTS

ILLUSTRATIONS

FREQFIT--COMPUTER PROGRAM WHICH PERFORMS NUMERICAL REGRESSION AND STATISTICAL CHI-SQUARED GOODNESS OF FIT ANALYSIS

by

Gregg S. Hofland and Christopher C. Barton

ABSTRACT

The computer program FREQFIT is designed to perform regression and statistical chi-squared goodness of fit analysis on one-dimensional or two-dimensional data. The program features an interactive user dialogue, numerous help messages, an option for screen or line printer output, and the flexibility to use practically any commercially available graphics package (such as LOTUS 1-2-3, GNUPLOT, or Golden Software's GRAPHER) to create plots of the program's results.

FREQFIT is written in Microsoft QuickBASIC, for IBM-PC compatible computers. A listing of the QuickBASIC source code for the FREQFIT program, a user manual, and sample input data, output, and plots are included.

INTRODUCTION

FREQFIT is a computer program written in Microsoft QuickBASIC version 3 (Appendix A), which performs numerical regression and statistical chi-squared goodness of fit analysis on one-dimensional and two-dimensional data. One-dimensional data typically consist of the frequency of a single characteristic (for example, the modal abundance of olivine or length of fractures). FREQFIT will read up to 8000 one-dimensional data points for analysis. Two-dimensional data consist of a set of X-Y data points. FREQFIT will read up to 500 two-dimensional data points for analysis.

If the original data set is one-dimensional, FREQFIT will create a data file containing a histogram with up to 500 cells. This histogram is statistically analyzed using chi-squared goodness of fit analysis. A data file containing a normal (gaussian) distribution curve for this histogram is then created. The midpoint of each of the histogram cells is then treated as a single data point of an X-Y graph and a data file is created containing these data points. From these X-Y points, the equation of best fit linear, exponential, logarithmic, and power regression curve is determined. For each form of regression analysis, a coefficient of determination is calculated, which indicates the quality of fit achieved by the curve. Coefficient values close to 1.00 indicate a closer fit than values close to 0.00 (see Appendix F). Data files are then created containing each of these regression curves. If the original data set is two-dimensional, FREQFIT will perform only the regression analysis. Figure 1 shows the program structure.

After FREQFIT has run, the data files created may be viewed or plotted using any commercial graphing package which reads ASCII data files, such as GNUPLOT, LOTUS 1-2-3, or Golden Software's GRAPHER. Graphs are made by importing the histogram or X-Y data file, and the optimum regression curve file. The two files can be overlaid, yielding a single plot of the raw data and the best fit curve through that data.

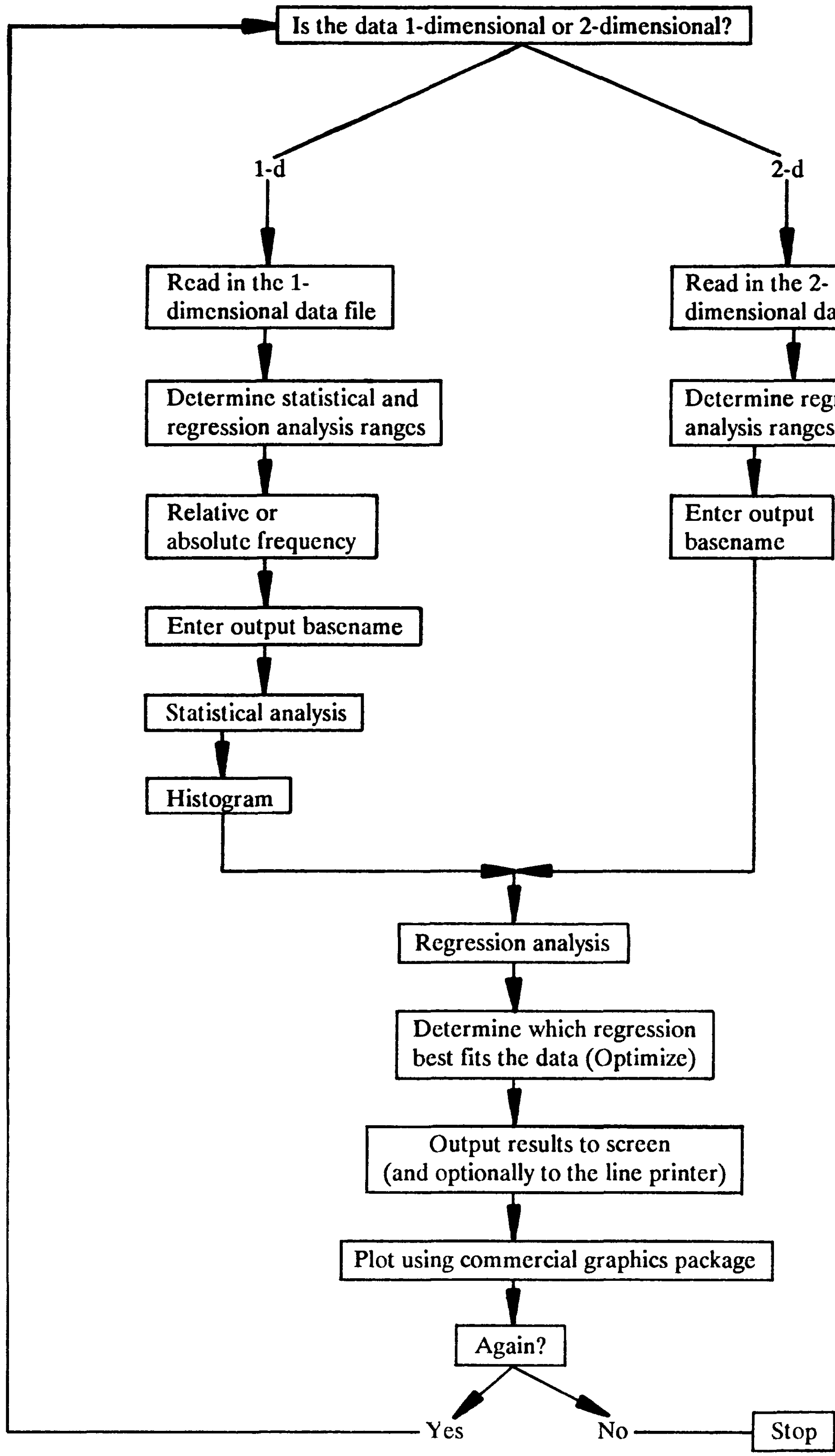


Figure l.--FREQFIT program structure

INPUT/OUTPUT FILES

The input file for FREQFIT may consist of either one-dimensional or two-dimensional data. In either case, the input data file should consist of numbers (either real or integer), separated by a comma or a carriage return, as shown in Appendices B and C.

After normal execution, FREQFIT will have created up to seven output data files:

One-Dimensional Output Data File:

xxxHIST.DAT - Histogram data file

xxxNORM.DAT - Gaussian distribution curve data file

xxxX-Y.DAT - X-Y curve data file

One or Two-Dimensional Output Data File:

xxxLIN.DAT - Linear regression curve data file

xxxEXP.DAT - Exponential regression curve data file

xxxLOG.DAT - Logarithmic regression curve data file

xxxPOW.DAT - Power regression curve data file

where xxx represents a file identification label, entered by the user.

Each of the output data files created by FREQFIT consist of a set of (X,Y) data points in ASCII format. For example:

1.05, 10.80

1.12, 10.75

1.15, 10.62

.

.

.

etc.

These files may be easily read by any commercially available graphics software package. Curves may be stacked on top of each other for comparison and further analysis by having the graphics software package plot data files simultaneously.

PROGRAM OPERATIONS

General Instructions

Most of the questions asked by FREQFIT are simple to understand and include defaulted answers. If you are confused about a question that lacks a default, press the carriage return key, and FREQFIT will give a short explanation of the information needed.

To execute FREQFIT, type 'freqfit' at the DOS prompt:

A:\> FREQFIT

The program will then ask you if the input data is one-dimensional or two-dimensional. Enter '1' if you have one-dimensional data (a group of measurements of a single characteristic), or enter '2' if your data consists of a set of X-Y points:

is your input data one-dimensional or two-dimensional? 1

FREQFIT will then ask for the name of the input file. Enter the directory path (if different from the current directory path) and the input file name:

enter input filename: A:EXAMPLE.DAT

This is the file which contains your original one- or two- dimensional data. This data file can be created with any ASCII text editor (such as EDLIN, Sidekick, or Microsoft Word), following the data file formats found in Appendices B and C.

One-Dimensional Data

If your data is one-dimensional, the program will read the input data file and determine the minimum and maximum ranges of your data. These values will be displayed, then the program will ask for the minimum and maximum histogram range desired. FREQFIT will take the entered values and construct a histogram from data found within this range including the endpoints. If you wish to run the analysis on only a part of the original data, enter the minimum and maximum ranges of the desired analysis. If you wish the program to consider the entire datafile, press the carriage return key at each of the defaulted prompts:

minimum data value = 1
maximum data value = 100

enter minimum analysis range: 1 [RETURN]
enter maximum analysis range: 100 [RETURN]

After you have entered the range of the histogram, the program also needs to know how many subdivisions (cells) within the histogram you desire. This information can be entered in two different ways: first, the program will ask you for the cell width. If you know how wide you want each cell, but haven't calculated the total number of cells between the minimum and maximum values, enter this width here. If, on the other hand, you know the total number of cells desired, but haven't calculated the width of an individual cell, press the carriage return key at the cell width prompt, and you will be prompted further for the total number of cells in the histogram. If you don't have any idea how wide you want each cell, or how many cells to use in the analysis, press the carriage return key again, and the program will determine the optimal cell width based on the data you have available (see appendix F). Remember that the number of cells within your histogram is also the number of X-Y data points to be used in the regression analysis. This entry, therefore, has a significant effect on your final results.

After entering the histogram size and cell width parameters, the program will ask whether you would prefer relative or absolute frequency. This entry has no effect on the actual results derived by the program, but rather changes the way the data are displayed. Relative frequency shows the percentage of the total population existing in each cell. Absolute frequency shows the true population values determined for each cell.

After all of this information has been entered, the program will create seven separate data files: a histogram, a normalized curve, an X-Y data set, and four best-fit regression curves. Before creating these files, the program will prompt you for a three letter identification label for the files. This label will be used to name the various files created. For example, if you entered ABC as your identification label, the program will create ABCHIST.DAT, ABCNORM.DAT, ABCX-Y.DAT, ABCLIN.DAT, ABCEXP.DAT, ABCLOG.DAT, and ABCPOW.DAT, corresponding to the histogram, gaussian, X-Y, linear regression, exponential regression, logarithmic regression, and power regression curves respectively.

Two-Dimensional Data

If your data is two-dimensional, FREQFIT will read the X and Y values from your input data file, and determine the maximum and minimum ranges of your data. These values will be displayed, then the program will ask for the minimum and maximum X-Y range desired for the regression analysis. If you wish to run the regression analysis on only a part of the original data, enter the minimum and maximum ranges of the desired analysis. Regression will be performed on the selected range, including the endpoints. If you wish the program to consider the entire data file, press the carriage return key at each of the defaulted prompts:

```
minimum data value =   1
maximum data value = 100

enter minimum analysis range:   1 [return]
enter maximum analysis range: 100 [return]
```

The program will then execute the regression analysis package on your data, creating linear, exponential, logarithmic, and power best-fit regression curves. Similar to the one-dimensional analysis case, you will be prompted for a three letter identification label which will be used to name the various files created.

Results of the Analysis

For both the one-dimensional and two-dimensional cases, after the output files are created, the program will output results of the analyses to the screen and optionally to the line printer. After these results are displayed, you will have the option of running FREQFIT again, or exiting to DOS.

<u>Examples of Program Execution</u>

One-Dimensional Data

Data in Appendix B

Histogram output in Appendix D

FREQFIT

is your input data one-dimensional or two-dimensional? 1

FREQFIT

enter input filename: A:EXAMPLE1.dat

FREQFIT

enter input filename: A:EXAMPLE1.DAT

minimum data value = 3
maximum data value = 20

enter minimum analysis range: 0
enter maximum analysis range: 20

cell width: 2

would you prefer relative frequency or absolute frequency? A

The program will now create a histogram file and best fit curve files. These files will contain data points which can be viewed or plotted after the program has executed.

enter a three letter identification label for files to be created: EX1

do you wish a paper copy of the results? Y

normalized curve overlay:

number of x-y points = 10
the mean is calculated to be = 10.98652
the variance is calculated to be = 13.47279
the standard deviation is calculated to be = 3.67053

chi-squared with 7 degrees of freedom = 39.45163
prob (chi-squared > 39.45163) = 0

HISTOGRAM X-Y DATA

X	Y
1.00	0.000
3.00	13.000
5.00	43.000
7.00	35.000
9.00	85.000
11.00	74.000
13.00	65.000
15.00	33.000
17.00	20.000
19.00	3.000

best curve fit using power regression:
Y = 4.093951 * X^ .8028849

coefficient of determination: .2554561

------------------------------------other solutions------------------------------------

coefficients of determination:
LIN = 4.227447E-03
EXP = 2.807905E-02
LOG = .1188087

equation:
Y = 33.91818 + 4.029713E-02 * X
Y = 13.96248 * exp (4.029713E-02 * X)
Y = 13.96248 + 4.029713E-02 * ln(X)

<u>Examples of Program Execution</u>
Two-Dimensional Data
Data in Appendix C
Histogram and Power Output in Appendix E

FREQFIT

is your input data one-dimensional of two-dimensional? 2

FREQFIT

enter input filename: EXAMPLE2.DAT

minimum data value = .5
maximum data value = 24.5

enter minimum analysis range: .5
enter maximum analysis range: 24.5

The program will now create best fit curve files. These files will contain data points which can be viewed or plotted after the program has executed.

enter a three letter identification label for files to be created: EX2

FREQFIT

normalized curve overlay:

number of X-Y points = 25
the mean is calculated to be = 1.70764
the variance is calculated to be = 5.026736
the standard deviation is calculated to be = 2.242038

chi-squared with 22 degrees of freedom = 1.035862E+20
prob (chi-squared > 1.035862E+20) = 0

HISTOGRAM X-Y DATA

X	Y
0.50	348.000
1.50	141.000
2.50	73.000
3.50	24.000
4.50	12.000
5.50	13.000
6.50	7.000
7.50	10.000
8.50	6.000
9.50	6.000
10.50	1.000
11.50	0.000
12.50	0.000
13.50	1.000
14.50	1.000
15.50	0.000
16.50	0.000
17.50	0.000
18.50	0.000
19.50	0.000
20.50	0.000
21.50	0.000
22.50	0.000
23.50	0.000
24.50	1.000

best curve fit using power regression:

Y = 192.169 * X^-1.797094

coefficient of determination: .9238803

-------------------------------------other solutions-----------------------------------

coefficients of determination:

LIN = .2865208
EXP = .7297467
LOG = .6968048

equation:

Y = 92.885 + -.2074881 * X
Y = 46.50941 * exp(-.2074881 * X)
Y = 46.50941 + -.2074881 * ln(X)

NOTES ABOUT IMPLEMENTATION

The FREQFIT program runs on IBM-PC, XT, and AT computers and compatibles. It requires a minimum of 256K RAM memory, and one floppy disk drive. Additional memory may be required when viewing the data using various graphics packages.

REFERENCES

Draper, N.R., 1966, Applied regression analysis: New York, John Wiley and Sons, 407 p.

Ezekiel, Mordecai, 1959, Methods of correlation and regression analysis, linear and curvilinear: New York, John Wiley and Sons, 548 p.

Hewlett-Packard, 1982, Statistical graphics for the HP-9845, part no. 09845-15121, 5th edition: Fort Collins, Colorado, Hewlett-Packard Desktop Computer Division, p. 131-133.

Hoaglin, D.C., and Welsch, R.E., 1978, The hat matrix in regression and ANOVA: The American Statistician, vol. 32, No. 1, p. 17-22.

Koetke, Walter, 1981, Multiple precision with micros: More multiple precision arithmetic: Kilobaud Microcomputing, v.5, no. 6, p. 20-23.

Freedman, David, Pisani, Robert, and Purves, Roger, 1979, Statistics: New York, W.W. Norton and Company, 506 p.

APPENDIX A -- FREQFIT program listing

```
'FREQFIT.BAS    V6.2      USGS//HOFLAND      20 August 1987 (revised 7/20/88)
'
'This program performs regression analysis on one-dimensional or two-
'dimensional data.  When analyzing one-dimensional data, FREQFIT creates
'a histogram with up to 500 cells.  This histogram is statistically analyzed
'to determine the mean, variance, and standard deviation of the data.  A
'normalized curve is then fit through the histogram, and the chi-squared
'goodness of fit is evaluated.  The vertical limit of each cell within the
'histogram creates an X-Y data set, which is curve fitted using linear,
'exponential, logarithmic, and power regression analysis.   For each form of
'regression, a coefficient of determination is calculated, which indicates
'the quality of fit achieved by the regression.  Coefficient values close
'to 1.00 indicate a closer fit than values close to 0.00.

'Note:  Regression analysis is based only on those data which do not give
'       errors during the calculations (ie. only positive data values are
'       used for logarithmic, exponential, and power curve fits).

'For the one-dimensional case, the program displays the mean, variance, and
'standard deviation of the data.  Results of the chi-squared goodness of fit
'analysis are then displayed for the normalized curve.  Then for either the
'one or two dimensional case, the X-Y data points are tabulated and the
'equation of the best fit regression curve is displayed, as well as its
'coefficient of determination.  Regression equations and coefficients of
'determination are also shown for the less optimal curve fits.

'After the program has run, results can be viewed or plotted using any
'commercial graphing package which reads ASCII data files, such as GNUPLOT
'or Golden Software's GRAPHER.  Graphs are made by importing the files
'???X-Y.DAT and the optimum regression curve file.  The two data files will
'overlay each other, giving a single plot of raw data and the best fit curve
'through that data.

'**  12/4/87  revisions:  Regression parameters were in error previously,
'                         now has been corrected.  Regression now assumes
'                         that zero amplitude values are not statistically
'                         significant.
'
'**  7/20/88              Bug fix for two dimensional data.


'***START***    FREQFIT.BAS     V6.2         USGS//HOFLAND      20 July 1988

DIM x(8000), y(500), observed.frequency(500), relative.observed.frequency(500),
contribution.chi.squared(500), d(500), expected.frequency(500), prob(21), valu(50), tempx(500),
tempy(500)

start:
CLEAR    'sets all variables to zero
COLOR 7, 0
CLS
pi = 3.141592654#
```

```
LOCATE 2, 2: PRINT "FREQFIT"

LOCATE 4, 5
input.start:
  INPUT "is your input data one-dimensional or two-dimensional?   ", data.dim


'ON ERROR GOTO error.select    'subroutine checks for disk errors

IF data.dim = 2 THEN
  CLS
  LOCATE 2, 2: PRINT "FREQFIT"
  LOCATE 5, 5: INPUT "enter input filename:  ", file1$
  OPEN file1$ FOR INPUT AS #1

  INPUT #1, x(1), y(1)
  min.x = x(1)
  max.x = min.x
  i = 2
  WHILE NOT EOF(1)
    INPUT #1, x(i), y(i)
    IF min.x > x(i) THEN min.x = x(i)
    IF max.x < x(i) THEN max.x = x(i)
    i = i + 1
  WEND


ELSEIF data.dim = 1 THEN
  CLS
  LOCATE 2, 2: PRINT "FREQFIT"
  LOCATE 5, 5: INPUT "enter input filename:  ", file1$
  OPEN file1$ FOR INPUT AS #1

input.first:              'null values in the data file are entered with a "-"
       INPUT #1, value$       'must check for the validity of each entry
       IF MID$(value$, 1, 1) <> "-" THEN
       x(1) = VAL(value$)
       min.x = x(1)
       max.x = min.x
     ELSE
       GOTO input.first
     END IF

    i = 1
    WHILE NOT EOF(1)
      i = i + 1
      INPUT #1, value$
      IF MID$(value$, 1, 1) <> "-" THEN
        x(i) = VAL(value$)
        IF max.x < x(i) THEN max.x = x(i)
        IF min.x > x(i) THEN min.x = x(i)
      END IF
    WEND
```

```
ELSE
  CLS
  PRINT : PRINT : PRINT
  PRINT "Your input data file can consist of single-dimensional data values"
  PRINT "(ie, a bunch of numbers separated by commas), or a two-dimensional"
  PRINT "data array (ie, a set of X-Y data points)."
  PRINT : PRINT
  GOTO input.start
END IF
CLOSE #1
points = i
PRINT

ranges:
  LOCATE 7, 5: PRINT "minimum data value = "; min.x
  LOCATE 8, 5: PRINT "maximum data value = "; max.x
  PRINT
  LOCATE 10, 5: PRINT "enter minimum analysis range:  "; min.x




'* subroutines min.cell, wowwie, max.cell, and woowie are used for inputting
'  the minimum and maximum histogram cell ranges.  The code found in these
'  subroutines was written to give the program a better appearance to the
'  user.  Subroutines min.cell and max.cell flash an underscore character
'  at the current input prompt, and check to see if a key was pressed by the
'  user.  Subroutines wowwie and woowie check to see if the key pressed was
'  a carriage return or a number (positive or negative).


min.cell:
     FOR i = 1 TO 50
        LOCATE 10, 36: PRINT "_";
        min.key$ = INKEY$
        IF LEN(min.key$) <> 0 THEN GOTO wowwie
     NEXT i
     FOR i = 1 TO 70
        LOCATE 10, 36: PRINT " ";
        min.key$ = INKEY$
        IF LEN(min.key$) <> 0 THEN GOTO wowwie
     NEXT i

wowwie:
     LOCATE 10, 36: PRINT " ";
     IF min.key$ <> CHR$(&HD) AND min.key$ <> CHR$(&H2D) AND (min.key$ < CHR$(&H30) OR min.key$ >
CHR$(&H39)) THEN GOTO min.cell           '0-9

     IF min.key$ = CHR$(&HD) THEN   'carriage return
        lowest.cell$ = ""


     ELSE
```

```
      LOCATE 10, 37: PRINT SPC(20);
      LOCATE 10, 37: PRINT min.key$;
      LOCATE 10, 38: INPUT "", lowest.cell$
      lowest.cell$ = min.key$ + lowest.cell$
    END IF

    LOCATE 10, 37: PRINT lowest.cell$;
    LOCATE 11, 5: PRINT "enter maximum analysis range:  "; max.x



max.cell:
    FOR i = 1 TO 50
       LOCATE 11, 36: PRINT "_";
       max.key$ = INKEY$
       IF LEN(max.key$) <> 0 THEN GOTO woowie
    NEXT i

    FOR i = 1 TO 70
       LOCATE 11, 36: PRINT " ";
       max.key$ = INKEY$
       IF LEN(max.key$) <> 0 THEN GOTO woowie
    NEXT i



woowie:
    LOCATE 11, 36: PRINT " ";
    IF max.key$ <> CHR$(&HD) AND max.key$ <> CHR$(&H2E) AND (max.key$ < CHR$(&H30) OR max.key$ >
CHR$(&H39)) THEN GOTO max.cell        '0-9

    IF max.key$ = CHR$(&HD) THEN
      highest.cell$ = ""

    ELSE
      LOCATE 11, 37: PRINT SPC(20);
      LOCATE 11, 37: PRINT max.key$;
      LOCATE 11, 38: INPUT "", highest.cell$
      highest.cell$ = max.key$ + highest.cell$
    END IF





'reorder the x and y arrays according to valid data ranges

IF LEN(lowest.cell$) > 0 THEN min.x = VAL(lowest.cell$)
IF LEN(highest.cell$) > 0 THEN max.x = VAL(highest.cell$)
IF data.dim = 2 THEN
  valid.points = 0
  FOR i = 1 TO points'determine number of data points within the range
    IF x(i) >= min.x AND x(i) <= max.x THEN
      valid.points = valid.points + 1          'number of valid points
```

```
      x(valid.points) = x(i)
      y(valid.points) = y(i)
    END IF
 NEXT i
END IF


IF data.dim = 1 THEN
    count = 0
    sum.x = 0
    sum.xsqr = 0

    FOR i = 1 TO points
      IF ((x(i) >= min.x) AND (x(i) <= max.x)) THEN
        count = count + 1
        sum.x = sum.x + x(i)
        sum.xsqr = sum.xsqr + x(i) ^ 2
      END IF
    NEXT i

    points = count

    IF points = 0 THEN
      PRINT : PRINT
      PRINT "error found in selecting cell ranges  -  try again."
      PRINT : PRINT "press any key to continue."
idleit:
       IF LEN(INKEY$) > 0 THEN GOTO start
      GOTO idleit
    END IF

    PRINT : PRINT
    y.now = CSRLIN
    x.now = POS(0)
    INPUT "    cell width:  ", cell$

    IF LEN(cell$) = 0 THEN
      LOCATE y.now, x.now: INPUT "    enter total number of cells:  ", cell$
      IF cell$ = "" THEN
        cells = INT(1 + 3.3 * LOG(points))
      ELSE
        cells = INT(VAL(cell$))
      END IF
      cell.width = (max.x - min.x) / cells
    ELSE
      cell.width = VAL(cell$)
      cells = INT((max.x - min.x) / cell.width)
    END IF
    remainder = (max.x - min.x) / cell.width - cells

    PRINT
    IF remainder > .01 OR remainder < -.01 THEN
      BEEP
      LOCATE 20, 5
```

```
      PRINT "cell width does not divide evenly into histogram range."
      PRINT
      PRINT "    new histogram range will be:  ("; min.x; ", "; max.x - cell.width * remainder; ")"
      INPUT "    is this range acceptable?  ", yorn$
      IF MID$(yorn$, 1, 1) = "n" OR MID$(yorn$, 1, 1) = "N" THEN
        CLS
        LOCATE 2, 2: PRINT "FREQFIT"
        LOCATE 5, 5: PRINT "enter input filename:  ", file1$
        GOTO ranges
      END IF
    max.x = max.x - remainder
    END IF


    CLS : PRINT : PRINT
    PRINT "    would you prefer relative freqency";
    INPUT " or absolute frequency?  ", plot.type$
    IF (MID$(plot.type$, 1, 1) <> "r" AND MID$(plot.type$, 1, 1) <> "R" AND MID$(plot.type$, 1, 1) <>
"A" AND MID$(plot.type$, 1, 1) <> "a") THEN
      PRINT
      PRINT "           A relative frequency histogram will show what percent"
      PRINT "        of the total population exists in each cell.  An absolute"
      PRINT "        frequency histogram will show the true population values "
      PRINT "        determined for each cell."
    END IF

recheck:
    IF (MID$(plot.type$, 1, 1) <> "r" AND MID$(plot.type$, 1, 1) <> "R" AND MID$(plot.type$, 1, 1) <>
"A" AND MID$(plot.type$, 1, 1) <> "a") THEN
      PRINT
      PRINT
      PRINT "           To select the type of histogram desired, enter an 'R'"
      PRINT "        for relative frequency, or enter an 'A' for absolute"
      PRINT "        frequency."
      PRINT
      INPUT "  Select histogram type:  ", plot.type$
      GOTO recheck
    END IF

    CLS : PRINT : PRINT
    PRINT "   The program will now create a histogram file and best fit"
    PRINT "   curve files.  These files will contain data points which can"
    PRINT "   be viewed or plotted after the program has executed."
    PRINT
    PRINT : PRINT
ELSE
    CLS : PRINT : PRINT
    PRINT "   The program will now create best fit curve files.  These files "
    PRINT "   will contain data points which can be viewed or plotted after"
    PRINT "   the program has executed."
    PRINT
    PRINT : PRINT
END IF
```

```
base.name:
  PRINT "  enter a three letter identification label";
  INPUT " for files to be created:  ", file2$
  IF LEN(file2$) = 0 THEN
    CLS
    PRINT : PRINT
    PRINT "   The program is asking for a name, which it will use to reference"
    PRINT "   created files.  When you enter the label, the computer will use"
    IF data.dim = 1 THEN
      PRINT "   this name to create seven separate files:  a histogram,"
      PRINT "   a normalized curve, an X-Y data set, and four regression curves."
      PRINT "   For example, if you entered 'ABC' for your identification label,"
      PRINT "   the program would create the files ABCHIST.DAT, ABCNORM.DAT"
      PRINT "   ABCX-Y.DAT, ABCLIN.DAT, ABCEXP.DAT, ABCLOG.DAT, and ABCPOW.DAT."
    ELSE
      PRINT "   this name to create four regression curves.  For example, if you"
      PRINT "   entered 'ABC' for your identification label, the program would"
      PRINT "   create the files ABCLIN.DAT, ABCEXP.DAT, ABCLOG.DAT, and ABCPOW.DAT."
    END IF
    PRINT
    PRINT "   After the calculations are complete, the computer will display"
    PRINT "   which regression method had the best fit.  Then, when you view"
    PRINT "   the data, select that regression file to overlay the X-Y data "
    PRINT "   set.  The other regression files, although less than optimum, "
    PRINT "   are still created in the event that you want to check the fit"
    PRINT "   of these curves over your data set."
    PRINT
    GOTO base.name
  END IF
IF data.dim = 1 THEN
    LOCATE 23, 1: COLOR 0, 7: PRINT "STATISTICS";
    xbar = sum.x / points
    variance = (sum.xsqr - ((sum.x ^ 2) / points)) / (points - 1)
    standard.deviation = SQR(variance)
    LOCATE 23, 12: PRINT "HISTOGRAM";

    FOR i = 1 TO points
      FOR j = 1 TO cells
         IF x(i) <= (min.x + cell.width * j) THEN
           observed.frequency(j) = observed.frequency(j) + 1
           j = cells
         END IF
      NEXT j
    NEXT i

    FOR i = 1 TO cells
       IF ((MID$(plot.type$, 1, 1) = "R") OR (MID$(plot.type$, 1, 1) = "r")) THEN
          relative.observed.frequency(i) = observed.frequency(i) * 100 / points
       ELSE
          relative.observed.frequency(i) = observed.frequency(i)
       END IF
       x(i) = min.x + (i - 1) * cell.width + cell.width / 2
       y(i) = relative.observed.frequency(i)
    NEXT i
```

```
  non.zero.pts = 0

  FOR i = 1 TO cells
    IF y(i) > 0 THEN
      non.zero.pts = non.zero.pts + 1
      tempy(non.zero.pts) = y(i)
      tempx(non.zero.pts) = x(i)
    END IF
  NEXT i
  FOR i = 1 TO non.zero.pts
    y(i) = tempy(i)
    x(i) = tempx(i)
  NEXT i


    'write histogram to disk file
    'write X-Y data set to disk file

    OPEN MID$(file2$, 1, 3) + "HIST.DAT" FOR OUTPUT AS #2
    FOR i = 1 TO cells
       PRINT #2, min.x + (i - 1) * cell.width, 0
       PRINT #2, min.x + (i - 1) * cell.width, relative.observed.frequency(i)
       PRINT #2, min.x + i * cell.width, relative.observed.frequency(i)
       PRINT #2, min.x + i * cell.width, 0
    NEXT i
    CLOSE #2

    OPEN MID$(file2$, 1, 3) + "X-Y.DAT" FOR OUTPUT AS #3
    FOR i = 1 TO non.zero.pts
      PRINT #3, x(i), y(i)
    NEXT i
    CLOSE #3


    'write normalized curve overlay to disk file
    LOCATE 23, 22: PRINT "NORMALIZED";
    OPEN MID$(file2$, 1, 3) + "NORM.DAT" FOR OUTPUT AS #3

    FOR i = xbar - 3.5 * standard.deviation TO xbar + 3.5 * standard.deviation STEP 7 *
standard.deviation / 128

      IF MID$(plot.type$, 1, 1) = "r" OR MID$(plot.type$, 1, 1) = "R" THEN
        nco = 100 * (cell.width) * EXP(((i - xbar) ^ 2) / (-2 * variance)) / SQR(2 * pi * variance)
      ELSE
        nco = points * (cell.width) * EXP(((i - xbar) ^ 2) / (-2 * variance)) / SQR(2 * pi * variance)
      END IF

      PRINT #3, i; ","; nco
    NEXT i

    CLOSE #3

    'determine chi squared values for normalized curve overlay
    chi.squared.df = 0
```

```
    FOR i = 1 TO cells   'determine expected frequency

      FOR j = min.x + cell.width * (i - 1) TO min.x + cell.width * i STEP .05
         co = points * cell.width * EXP(((j - xbar) ^ 2) / (-2 * variance)) / SQR(2 * pi * variance)
         expected.frequency(i) = expected.frequency(i) + co * .05 / cell.width
      NEXT j

      contribution.chi.squared(i) = (((observed.frequency(i) - expected.frequency(i)) ^ 2) /
expected.frequency(i))
      chi.squared.df = chi.squared.df + contribution.chi.squared(i)

    NEXT i

    df = cells - 3
    IF cells < 3 THEN
      BEEP
      LOCATE 23, 23: PRINT "*invalid*"
      norm.curve.flag = 1 'true if can't do normalized curve analysis
      GOTO 3890     'regression analysis routines
    END IF
    GOSUB calculate.exponential   'calculates y^x for very large x

    gamma = 1: prob = 0
    IF df + 10 * SQR(2 * df) < chi.squared.df THEN 3890
    IF df = 1 THEN 3750

    FOR i = (df + 2) / 2 - 1 TO 1.5 STEP -1
      gamma = gamma * i
    NEXT i

3750 IF df MOD 2 = 1 THEN gamma = gamma * .886226925#
    i = 0
    i0 = 0
    j = 0

3770 j = j + 1
    k = 1

    FOR l = 1 TO j
      k = k * (df + 2 * l)
    NEXT l

    i = i + valu(j) / k
    IF (ABS(i - i0) / i) < .000001 OR j = 30 THEN 3860
    i0 = i
    GOTO 3770
3860 prob = (1 - (chi.squared.df / 2) ^ (df / 2) * EXP(-chi.squared.df / 2) / gamma * (1 + i))

END IF

'execute regression analysis on X-Y data
 'set up parameters
```

```
3890 LOCATE 23, 33: COLOR 0, 7: PRINT "REGRESSION";
sum.x = 0
sum.y = 0
sum.xy = 0
sum.xsqr = 0
sum.ysqr = 0
sum.lnx = 0
sum.lny = 0
sum.lnxlny = 0
sum.lnysqr = 0
sum.lnxsqr = 0
sum.xlny = 0
sum.ylnx = 0

IF data.dim = 2 THEN
  non.zero.pts = 0
  FOR i = 1 TO valid.points
    IF y(i) > 0 THEN
      non.zero.pts = non.zero.pts + 1
      tempy(non.zero.pts) = y(i)
      tempx(non.zero.pts) = x(i)
    END IF
  NEXT i
  FOR i = 1 TO non.zero.pts
    y(i) = tempy(i)
    x(i) = tempx(i)
  NEXT i
END IF

FOR i = 1 TO non.zero.pts
   sum.x = sum.x + x(i)
   sum.y = sum.y + y(i)
   sum.xy = sum.xy + x(i) * y(i)
   sum.xsqr = sum.xsqr + (x(i)) ^ 2
   sum.ysqr = sum.ysqr + (y(i)) ^ 2
   sum.lnx = sum.lnx + LOG(x(i))
   sum.lnxsqr = sum.lnxsqr + (LOG(x(i))) ^ 2
   sum.ylnx = sum.ylnx + y(i) * LOG(x(i))
   sum.lny = sum.lny + LOG(y(i))
   sum.lnysqr = sum.lnysqr + (LOG(y(i))) ^ 2
   sum.xlny = sum.xlny + x(i) * LOG(y(i))
   sum.lnxlny = sum.lnxlny + LOG(x(i)) * LOG(y(i))
NEXT i



'linear regression   :   y = bx + a

  b1.num = sum.xy
  b2.num = sum.x * sum.y
  b1.den = sum.xsqr
  b2.den = sum.x ^ 2

  a1.num = sum.y
```

```
  a2.num = sum.x

  r1.num = sum.xy
  r2.num = sum.x * sum.y
  r1.den = sum.xsqr
  r2.den = sum.x ^ 2
  r3.den = sum.ysqr
  r4.den = sum.y ^ 2

  GOSUB regres     'does the actual regression for these parameters

  IF regres.error = 1 THEN
   lin.err = 1   'error determining linear regression
   regres.error = 0
  ELSE
   b.lin = b
   a.lin = a
   r.lin = r
  END IF



'exponential regression  :  y = a exp (bx)

  b1.num = sum.xlny
  b2.num = sum.x * sum.lny
  b1.den = sum.xsqr
  b2.den = sum.x ^ 2

  a1.num = sum.lny
  a2.num = sum.x


  r1.num = sum.xlny
  r2.num = sum.x * sum.lny
  r1.den = sum.xsqr
  r2.den = sum.x ^ 2
  r3.den = sum.lnysqr
  r4.den = sum.lny ^ 2

  GOSUB regres

  IF regres.error = 1 THEN
   exp.err = 1   'error determining exponental regression
   regres.error = 0
  ELSE
   b.exp = b
   a.exp = EXP(a)
   r.exp = r
  END IF



'logarithmic regression  :  y = a + b ln(x)
```

```
  b1.num = sum.ylnx
  b2.num = sum.lnx * sum.y
  b1.den = sum.lnxsqr
  b2.den = sum.lnx ^ 2

  a1.num = sum.y
  a2.num = sum.lnx


  r1.num = sum.ylnx
  r2.num = sum.lnx * sum.y
  r1.den = sum.lnxsqr
  r2.den = sum.lnx ^ 2
  r3.den = sum.ysqr
  r4.den = sum.y ^ 2

  GOSUB regres

  IF regres.error = 1 THEN
   log.err = 1   'error determining logarithmic regression
   regres.error = 0
  ELSE
   b.log = b
   a.log = a
   r.log = r
  END IF

'power regression  :  y = a x ^b

  b1.num = sum.lnxlny
  b2.num = sum.lnx * sum.lny
  b1.den = sum.lnxsqr
  b2.den = sum.lnx ^ 2

  a1.num = sum.lny
  a2.num = sum.lnx


  r1.num = sum.lnxlny
  r2.num = sum.lnx * sum.lny
  r1.den = sum.lnxsqr
  r2.den = sum.lnx ^ 2
  r3.den = sum.lnysqr
  r4.den = sum.lny ^ 2

  GOSUB regres

  IF regres.error = 1 THEN
   pow.err = 1   'error determining power regression
   regres.error = 0
  ELSE
   b.pow = b
   a.pow = EXP(a)
```

```
   r.pow = r
  END IF


'determine which regression methods were valid

  IF lin.err = 1 THEN r.lin = 0
  IF exp.err = 1 THEN r.exp = 0
  IF pow.err = 1 THEN r.pow = 0
  IF log.err = 1 THEN r.log = 0
  IF r.lin = 0 AND r.exp = 0 AND r.pow = 0 AND r.log = 0 THEN
    total.botch = 1   'all regression methods were invalid
    GOTO print.results
  END IF


'determine the optimum curve
  LOCATE 23, 44: PRINT "OPTIMIZE";

  optimal = 1
  r.opt = r.lin
  a.opt = a.lin
  b.opt = b.lin
  opt.name$ = "linear"
  opt.form1$ = "  +  "
  opt.form2$ = " *  X"

  IF r.opt < r.exp THEN
    optimal = 2
    r.opt = r.exp
    a.opt = a.exp
    b.opt = b.exp
    opt.name$ = "exponential"
    opt.form1$ = " * exp( "
    opt.form2$ = "  * X )"
  END IF

  IF r.opt < r.log THEN
    optimal = 3
    r.opt = r.log
    a.opt = a.log
    b.opt = b.log
    opt.name$ = "logarithmic"
    opt.form1$ = "  +  "
    opt.form2$ = " * ln (X)"
  END IF

  IF r.opt < r.pow THEN
    optimal = 4
    r.opt = r.pow
    a.opt = a.pow
    b.opt = b.pow
    opt.name$ = "power"
    opt.form1$ = " *  X^"
```

```
    opt.form2$ = ""
  END IF


'write curve overlay files to disk

  'linear regression file
   IF lin.err <> 1 THEN
     LOCATE 23, 53: PRINT "WRITING:  ";
     LOCATE 23, 63: PRINT MID$(file2$, 1, 3) + "LIN.DAT"
     OPEN MID$(file2$, 1, 3) + "LIN.DAT" FOR OUTPUT AS #2
     FOR i = min.x TO max.x STEP (max.x - min.x) / 200


       y.lin = a.lin + b.lin * i
       PRINT #2, i, y.lin

     NEXT i
     CLOSE #2
   END IF

  'exponential regression file
   IF exp.err <> 1 THEN
     LOCATE 23, 53: PRINT "WRITING:  ";
     LOCATE 23, 63: PRINT MID$(file2$, 1, 3) + "EXP.DAT"
     OPEN MID$(file2$, 1, 3) + "EXP.DAT" FOR OUTPUT AS #2
     FOR i = min.x TO max.x STEP (max.x - min.x) / 200
       y.exp = a.exp * EXP(b.exp * i)
       PRINT #2, i, y.exp
     NEXT i
     CLOSE #2
   END IF


  'logarithmic regression file
  IF log.err <> 1 THEN
    LOCATE 23, 53: PRINT "WRITING:  ";
    LOCATE 23, 63: PRINT MID$(file2$, 1, 3) + "LOG.DAT"
    OPEN MID$(file2$, 1, 3) + "LOG.DAT" FOR OUTPUT AS #2
    FOR i = min.x TO max.x STEP (max.x - min.x) / 200
       IF i > 0 THEN
         y.log = a.log + b.log * LOG(i)
         PRINT #2, i, y.log
       END IF
    NEXT i
    CLOSE #2
  END IF

  'power regression file
  IF pow.err <> 1 THEN
    LOCATE 23, 53: PRINT "WRITING:  ";
    LOCATE 23, 63: PRINT MID$(file2$, 1, 3) + "POW.DAT"
    OPEN MID$(file2$, 1, 3) + "POW.DAT" FOR OUTPUT AS #2
    FOR i = min.x TO max.x STEP (max.x - min.x) / 200
```

```
      IF i > 0 THEN
        y.pow = a.pow * i ^ b.pow
        PRINT #2, i, y.pow
      END IF
    NEXT i
    CLOSE #2
  END IF


print.results:
COLOR 7, 0
CLS
LOCATE 15, 5: PRINT "do you wish a papar copy of the results?  Y";
LOCATE 15, 47: INPUT "", hcopy$

IF MID$(hcopy$, 1, 1) = "y" OR MID$(hcopy$, 1, 1) = "Y" OR LEN(hcopy$) = 0 THEN
  LPRINT : LPRINT : LPRINT : LPRINT "FREQFIT": LPRINT : LPRINT file1$: LPRINT : LPRINT

  IF data.dim = 1 THEN
    LPRINT "normalized curve overlay:"
    LPRINT
    LPRINT "  number of X-Y points = "; cells
    LPRINT "  the mean is calculated to be = "; xbar
    LPRINT "  the variance is calculated to be = "; variance
    LPRINT "  the standard deviation is calculated to be = "; standard.deviation
    LPRINT : LPRINT

    IF norm.curve.flag = 1 THEN
      LPRINT : LPRINT "                    ** ERROR DETECTED **"
      LPRINT "insufficient number of cells for chi squared calculations."
      LPRINT "must have at least 3 cells for this type of analysis."
    ELSE
      LPRINT
      LPRINT "chi-squared with "; df; " degrees of freedom = "; chi.squared.df
      LPRINT "prob (chi-squared > "; chi.squared.df; ") = "; prob
    END IF
    LPRINT : LPRINT : LPRINT
  END IF
  LPRINT "______________________________________________________"
  LPRINT "X-Y DATA"
  LPRINT
  LPRINT "          X                 Y"
  LPRINT
  FOR i = 1 TO non.zero.pts
     LPRINT USING "  #######.##       #######.###  "; x(i), y(i)
  NEXT i
  IF total.botch = 1 THEN
    LPRINT
    LPRINT "**ERROR** "
    LPRINT "all forms of regression were invalid."
  ELSE
    LPRINT : LPRINT
    LPRINT "best curve fit using "; opt.name$; " regression:"
    LPRINT "     Y = "; a.opt; opt.form1$; b.opt; opt.form2$
```

```
LPRINT
LPRINT "coefficient of determination:  "; r.opt
LPRINT : LPRINT
LPRINT "------------------------ other solutions ------------------------"
LPRINT "coefficients of determination:           equation:"
IF optimal = 1 THEN

  IF exp.err = 1 THEN
    LPRINT " exponential regression method invalid."
  ELSE
    LPRINT "     EXP = "; r.exp, "              Y = "; a.exp; " * exp("; b.exp; " * X)"
  END IF

  IF log.err = 1 THEN
    LPRINT " logarithmic regression method invalid."
  ELSE
    LPRINT "     LOG = "; r.log, "              Y = "; a.log; " + "; b.log; " * ln(X)"
  END IF

  IF pow.err = 1 THEN
    LPRINT " power regression method invalid."
  ELSE
    LPRINT "     POW = "; r.pow, "              Y = "; a.pow; " * X ^ "; b.pow
  END IF


ELSEIF optimal = 2 THEN
  IF lin.err = 1 THEN
    LPRINT " linear regression method invalid."
  ELSE
    LPRINT "     LIN = "; r.lin, "              Y = "; a.lin; " + "; b.lin; " * X"
  END IF

  IF log.err = 1 THEN
    LPRINT " logarithmic regression method invalid."
  ELSE
    LPRINT "     LOG = "; r.log, "              Y = "; a.log; " + "; b.log; " * ln(X)"
  END IF

  IF pow.err = 1 THEN
    LPRINT " power regression method invalid."
  ELSE
    LPRINT "     POW = "; r.pow, "              Y = "; a.pow; " * X ^ "; b.pow
  END IF

ELSEIF optimal = 3 THEN
  IF lin.err = 1 THEN
    LPRINT " linear regression method invalid."
  ELSE
    LPRINT "     LIN = "; r.lin, "              Y = "; a.lin; " + "; b.lin; " * X"
  END IF

  IF exp.err = 1 THEN
    LPRINT " exponential regression method invalid."
```

```
    ELSE
      LPRINT "     EXP = "; r.exp, "             Y = "; a.exp; " * exp("; b.exp; " * X)"
    END IF

    IF pow.err = 1 THEN
      LPRINT " power regression method invalid."
    ELSE
      LPRINT "     POW = "; r.pow, "             Y = "; a.pow; " * X ^ "; b.pow
    END IF


   ELSEIF optimal = 4 THEN
    IF lin.err = 1 THEN
      LPRINT " linear regression method invalid."
    ELSE
      LPRINT "     LIN = "; r.lin, "             Y = "; a.lin; " + "; b.lin; " * X"
    END IF

    IF exp.err = 1 THEN
      LPRINT " exponential regression method invalid."
    ELSE
      LPRINT "     EXP = "; r.exp, "             Y = "; a.exp; " * exp("; b.exp; " * X)"
    END IF

    IF log.err = 1 THEN
      LPRINT " logarithmic regression method invalid."
    ELSE
      LPRINT "     LOG = "; r.log, "             Y = "; a.log; " + "; b.log; " * ln(X)"
    END IF
   END IF
  END IF
LPRINT : LPRINT : LPRINT : LPRINT : LPRINT : LPRINT
END IF



CLS
PRINT file1$
IF data.dim = 1 THEN
  PRINT "normalized curve overlay:"
  PRINT : PRINT
  PRINT "number of X-Y points = "; cells
  PRINT "the mean is calculated to be = "; xbar
  PRINT "the variance is calculated to be = "; variance
  PRINT "the standard deviation is calculated to be = "; standard.deviation
  IF norm.curve.flag = 1 THEN
    PRINT : PRINT "                  ** ERROR DETECTED **"
    PRINT "insufficient number of cells for chi squared calculations."
    PRINT "must have at least 3 cells for this type of analysis."
  ELSE
    PRINT
    PRINT "chi-squared with "; df; " degrees of freedom = "; chi.squared.df
    PRINT "prob (chi-squared > "; chi.squared.df; ") = "; prob
  END IF
```

```
  LOCATE 23, 5: PRINT "press any key to continue."
idle.idle:
  IF LEN(INKEY$) = 0 THEN GOTO idle.idle
END IF
CLS
PRINT "________________________________________________________"
PRINT "X-Y DATA:  "; file1$
PRINT
PRINT "          X                 Y"
PRINT
line.count = 6
line.orig = line.count

FOR i = 1 TO non.zero.pts
   x.now = POS(0)
   PRINT USING "     #######.##        #######.###  "; x(i), y(i)
   line.count = line.count + 1
   IF line.count = 22 AND i < non.zero.pts THEN
     LOCATE 23, 5: PRINT "press any key to continue."
idle12:
       IF INKEY$ <> "" THEN GOTO print.more
     GOTO idle12
print.more:
     LOCATE line.orig, x.now
     line.count = 6
   END IF
NEXT i
FOR i = line.count TO 22
   PRINT SPC(77);
NEXT i

LOCATE 23, 5: PRINT "press any key to continue."
idle13:
  IF INKEY$ <> "" THEN GOTO more.more
GOTO idle13

more.more:
  CLS
  PRINT file1$
  PRINT : PRINT
  IF total.botch = 1 THEN
    PRINT "**ERROR**"
    PRINT "all forms of regression were considered invalid."
    PRINT
  ELSE
    PRINT "best curve fit using "; opt.name$; " regression:"
    PRINT "     Y = "; a.opt; opt.form1$; b.opt; opt.form2$
    PRINT
    PRINT "coefficient of determination:  "; r.opt
    PRINT : PRINT
    PRINT "------------------------ other solutions ------------------------"
    PRINT "coefficients of determination:             equation:"
    IF optimal = 1 THEN
      IF exp.err = 1 THEN
```

```
    PRINT " exponential regression method invalid."
  ELSE
    PRINT "     EXP = "; r.exp, "              Y = "; a.exp; " * exp("; b.exp; " * X)"
  END IF

  IF log.err = 1 THEN
    PRINT " logarithmic regression method invalid."
  ELSE
    PRINT "     LOG = "; r.log, "              Y = "; a.log; " + "; b.log; " * ln(X)"
  END IF

  IF pow.err = 1 THEN
    PRINT " power regression method invalid."
  ELSE
    PRINT "     POW = "; r.pow, "              Y = "; a.pow; " * X ^ "; b.pow
  END IF


ELSEIF optimal = 2 THEN
  IF lin.err = 1 THEN
    PRINT " linear regression method invalid."
  ELSE
    PRINT "     LIN = "; r.lin, "              Y = "; a.lin; " + "; b.lin; " * X"
  END IF

  IF log.err = 1 THEN
    PRINT " logarithmic regression method invalid."
  ELSE
    PRINT "     LOG = "; r.log, "              Y = "; a.log; " + "; b.log; " * ln(X)"
  END IF

  IF pow.err = 1 THEN
    PRINT " power regression method invalid."
  ELSE
    PRINT "     POW = "; r.pow, "              Y = "; a.pow; " * X ^ "; b.pow
  END IF

ELSEIF optimal = 3 THEN
  IF lin.err = 1 THEN
    PRINT " linear regression method invalid."
  ELSE
    PRINT "     LIN = "; r.lin, "              Y = "; a.lin; " + "; b.lin; " * X"
  END IF

  IF exp.err = 1 THEN
    PRINT " exponential regression method invalid."
  ELSE
    PRINT "     EXP = "; r.exp, "              Y = "; a.exp; " * exp("; b.exp; " * X)"
  END IF

  IF pow.err = 1 THEN
    PRINT " power regression method invalid."
  ELSE
    PRINT "     POW = "; r.pow, "              Y = "; a.pow; " * X ^ "; b.pow
```

```
      END IF


    ELSEIF optimal = 4 THEN
      IF lin.err = 1 THEN
        PRINT " linear regression method invalid."
      ELSE
        PRINT "     LIN = "; r.lin, "            Y = "; a.lin; " + "; b.lin; " * X"
      END IF

      IF exp.err = 1 THEN
        PRINT " exponential regression method invalid."
      ELSE
        PRINT "     EXP = "; r.exp, "            Y = "; a.exp; " * exp("; b.exp; " * X)"
      END IF

      IF log.err = 1 THEN
        PRINT " logarithmic regression method invalid."
      ELSE
        PRINT "     LOG = "; r.log, "            Y = "; a.log; " + "; b.log; " * ln(X)"
      END IF
    END IF
   END IF
  PRINT : PRINT : PRINT
  PRINT "would you like to run the program again?  N"
  LOCATE CSRLIN - 1, 43: INPUT "", re.run$
  IF MID$(re.run$, 1, 1) = "n" OR MID$(re.run$, 1, 1) = "N" OR LEN(re.run$) = 0 THEN GOTO end.program
  GOTO start


error.select:
  IF ERR = 53 THEN
    PRINT : PRINT
    PRINT "file not found  -  press any key to continue."
idleerror:
      IF LEN(INKEY$) > 0 THEN RESUME start
    GOTO idleerror
  END IF

  PRINT "error found  -  press any key to continue."
  GOTO idleerror



regres:
  a = 0
  b = 0
  r = 0
  IF (b1.den - b2.den / non.zero.pts) = 0 OR non.zero.pts = 0 OR ((r1.den - r2.den / non.zero.pts) *
(r3.den - r4.den / non.zero.pts)) = 0 THEN
      regres.error = 1

  ELSE
     b = (b1.num - b2.num / non.zero.pts) / (b1.den - b2.den / non.zero.pts)
     a = (a1.num / non.zero.pts - b * a2.num / non.zero.pts)
```

```
    r = ((r1.num - r2.num / non.zero.pts) ^ 2) / ((r1.den - r2.den / non.zero.pts) * (r3.den - r4.den /
non.zero.pts))
  END IF
RETURN


calculate.exponential:
m = chi.squared.df

IF m > 18.8 THEN
  prob = 0
  RETURN 3890
END IF

n = 1: d(1) = 1

FOR k = 1 TO 30
  c = 0

  FOR i = 1 TO n
    p = d(i) * m + c
    c = p / 10
    d(i) = p - c * 10
  NEXT i

  d(n + 1) = c
  IF c > 0 THEN n = n + 1

  FOR i = n TO 1 STEP -1
      valu(k) = valu(k) + d(i) * 10 ^ (i - 1)
  NEXT i

NEXT k
RETURN



end.program:
  CLS
  END

'***END***   FREQFIT.BAS
```

APPENDIX B - One-Dimensional Input Example Data File (EXAMPLE1.DAT)

X	X--cont.	X--cont.
6	13	16
19	15	12
14	14	8
8	15	6
9	16	15
14	10	16
5	6	14
10	5	14
11	13	12
12	10	11
4	14	18
9	4	14
9	9	4
8	14	9
13	12	12
5	13	6
10	10	14
14	14	12
13	14	12
13	18	16
12	16	16
9	12	12
14	17	10
14	17	13
6	16	6
12	12	10
14	14	13
12	16	14
9	14	6
6	16	5
9	12	5
10	14	6
10	6	16
5	8	11
12	9	17
14	18	10
11	16	11
6	18	8
14	17	10
8	14	8
8	3	9
5	10	10
12	13	9
14	5	10
16	9	6
14	8	12
5	10	12
12	14	4
4	14	14
12	8	12
9	18	12
9	6	12

X--cont.	X--cont.	X--cont.
9	3	8
7	4	12
12	12	10
16	11	10
6	8	16
12	8	10
12	9	16
16	4	16
14	7	12
12	14	18
14	14	14
12	10	20
15	18	10
17	6	8
16	12	9
14	9	12
14	8	12
12	13	14
9	10	18
12	12	10
12	10	12
5	14	10
3	10	9
16	6	10
9	6	12
5	7	13
6	14	10
9	13	9
4	18	14
8	9	5
9	6	12
12	8	14
7	12	14
9	19	14
10	8	15
10	6	11
5	10	13
7	8	10
12	8	7
10	9	5
6	10	9
8	9	10
8	18	9
7	10	12
10	12	11
8	10	10
8	11	12
12	12	12
8	10	16
9	14	14
14	12	16
7	12	10

X--cont.	X--cont.
17	10
14	12
16	14
11	12
12	10
12	14
12	14
16	12
18	6
17	5
14	12
16	5
14	4
12	5
5	8
10	3
10	10
6	14
10	9
10	18
11	10
5	12
15	10
15	16
9	12
11	10
	5

APPENDIX C--Example Two-Dimensional Input Data File (EXAMPLE2.DAT)

X	Y
1.5	141
2.5	73
3.5	24
4.5	12
5.5	13
6.5	7
7.5	10
8.5	6
9.5	6
10.5	1
13.5	1
14.5	1
24.5	1

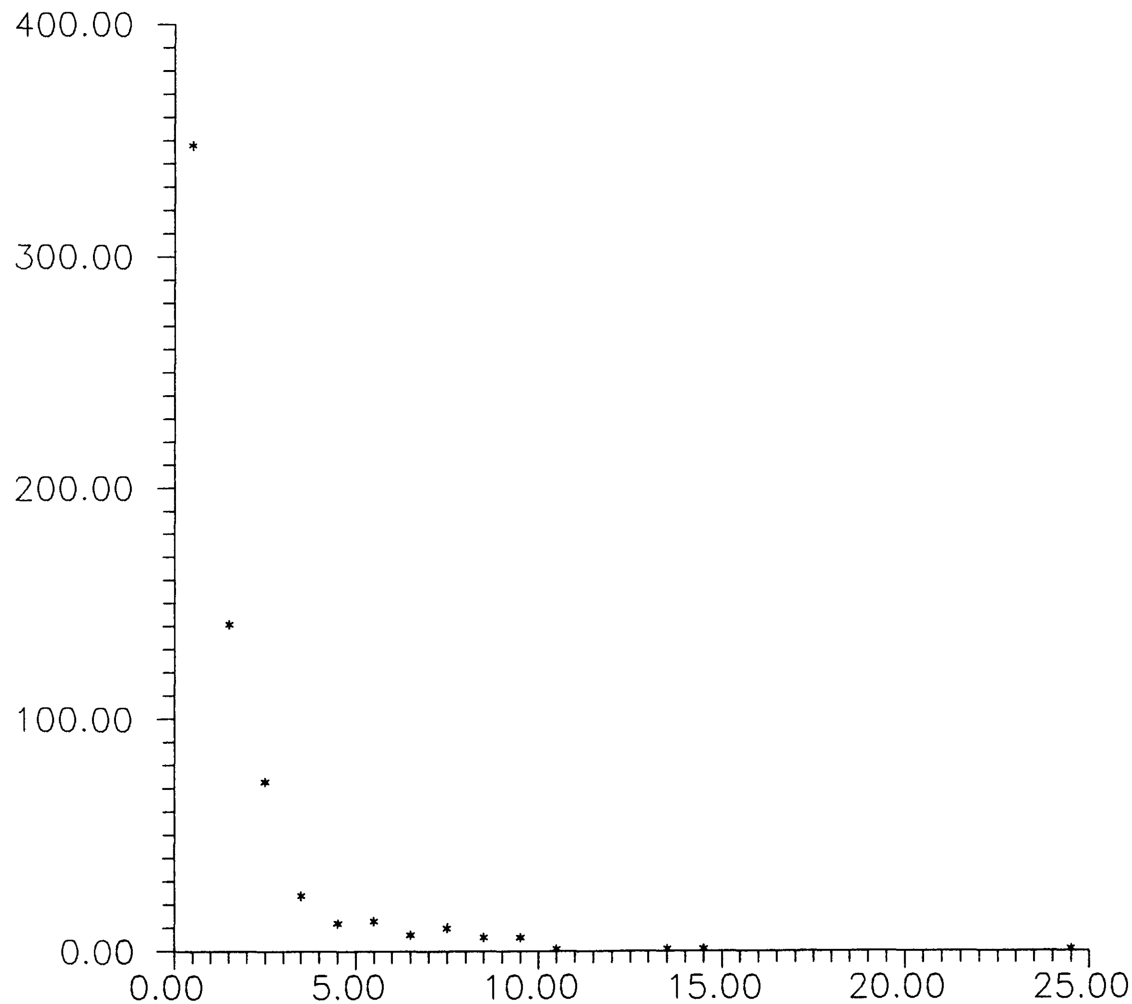


FIGURE C1. -- Plot of Two-Dimensional Input Data File (EXAMPLE2.DAT)

APPENDIX D--FREQFIT generated histogram output file (EX1HIST.DAT)

X	Y
0	0
2	0
2	0
2	0
2	13
4	13
4	0
4	0
4	43
6	43
6	0
6	0
6	35
8	35
8	0
8	0
8	85
10	85
10	0
10	0
10	74
12	74
12	0
12	0
12	65
14	65
14	0
14	0
14	33
16	33
16	0
16	0
16	20
18	20
18	0
18	0
18	3
20	3
20	0

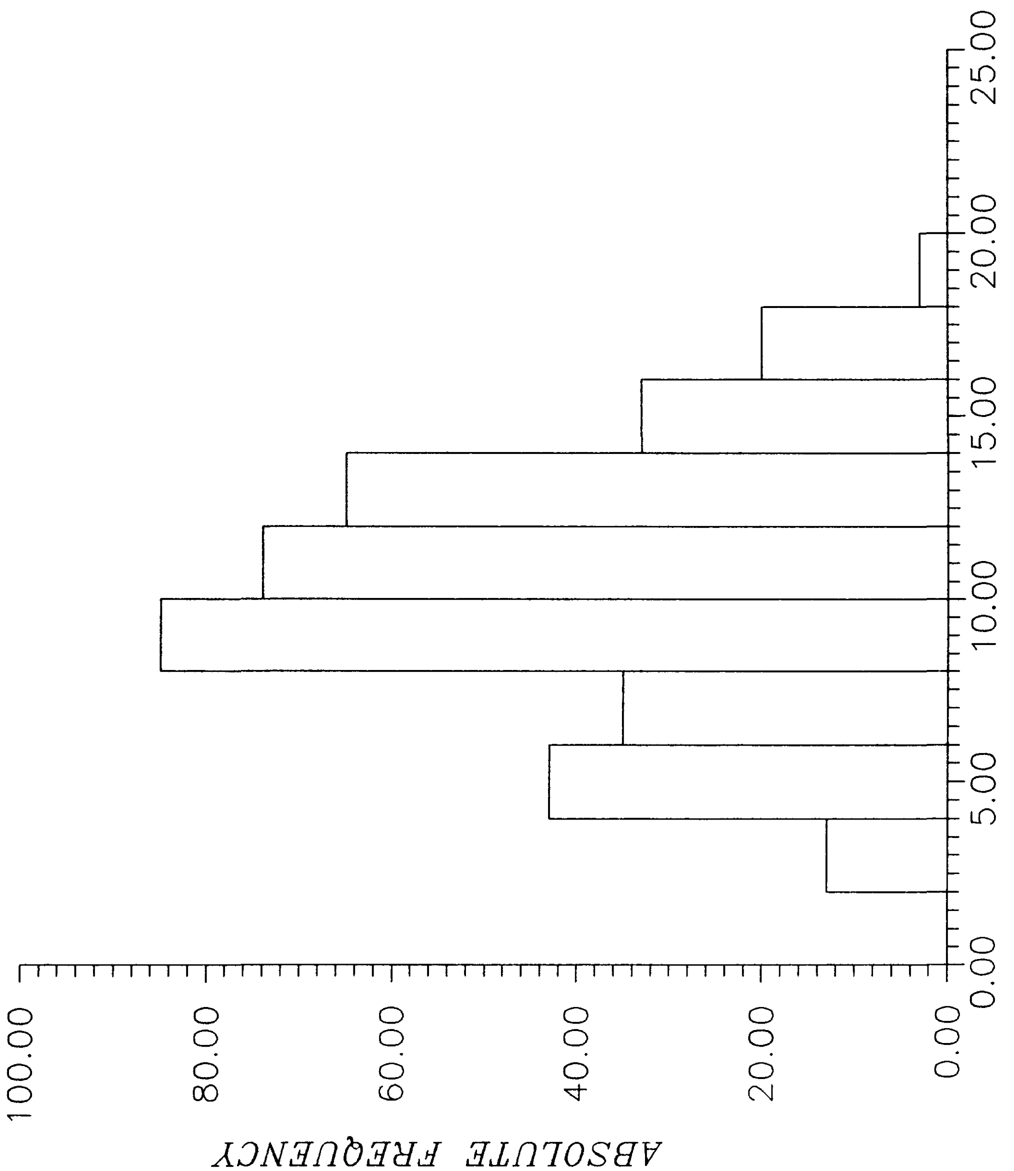


FIGURE D1.--Plot of FREQFIT generated histogram output file (EX1HIST.DAT)

APPENDIX E--FREQFIT generated best-fit curve output file (EX2POW.DAT)

X	Y
.62	472.1862
.74	344.4762
.86	263.5316
.98	208.7976
1.1	169.9465
1.22	141.3078
1.34	119.5488
1.46	102.6023
1.58	89.12829
1.7	78.22618
1.82	69.2716
1.94	61.82024
2.06	55.54865
2.18	50.21661
2.3	45.64259
2.42	41.68724
2.539999	38.24209
2.659999	35.22165
2.779999	32.55777
2.899999	30.19558
3.019999	28.09044
3.139999	26.20576
3.259999	24.5113
3.379999	22.98187
3.499999	21.59637
3.619998	20.33698
3.739998	19.1886
3.859998	18.13834
3.979998	17.17513
4.099998	16.28946
4.219998	15.47308
4.339998	14.71881
4.459998	14.02042
4.579998	13.37241
4.699997	12.76997
4.819997	12.20885
4.939997	11.6853
5.059997	11.19597
5.179997	10.73789
5.299997	10.30841
5.419997.	9.905156
5.539997	9.525999
5.659997	9.169022
5.779996	8.832498
5.899996	8.514864
6.019996	8.214709
6.139996	7.930751
6.259996	7.661824
6.379996	7.406866
6.499996	7.164909
6.619996	6.935069

X	Y
6.739995	6.716534
6.859995	6.508563
6.979995	6.310472
7.099995	6.121634
7.219995	5.941473
7.339995	5.769454
7.459995	5.605085
7.579995	5.447911
7.699995	5.29751
7.819994	5.153492
8.059995	4.88318
8.179995	4.756235
8.299994	4.634367
8.419994	4.517305
8.539994	4.404794
8.659994	4.296598
8.779994	4.192493
8.899994	4.092274
9.019994	3.995745
9.139994	3.902724
9.259994	3.81304
9.379993	3.726532
9.499993	3.643049
9.619993	3.562449
9.739993	3.484599
9.859993	3.409373
9.979993	3.336652
10.09999	3.266323
10.21999	3.198281
10.33999	3.132426
10.45999	3.068664
10.57999	3.006905
10.69999	2.947065
10.81999	2.889063
10.93999	2.832824
11.05999	2.778275
11.17999	2.725349
11.29999	2.67398
11.41999	2.624106
11.53999	2.57567
11.65999	2.528614
11.77999	2.482888
11.89999	2.438439
12.01999	2.39522
12.13999	2.353185
12.25999	2.31229
12.37999	2.272494
12.49999	2.233757
12.61999	2.196041
12.73999	2.15931
12.85999	2.123529

X	Y
12.97999	2.088665
13.09999	2.054686
13.21999	2.021562
13.33999	1.989265
13.45999	1.957766
13.57999	1.927039
13.69999	1.897058
13.81999	1.867798
13.93999	1.839238
14.05999	1.811353
14.17999	1.784122
14.29999	1.757526
14.41999	1.731543
14.53999	1.706154
14.65999	1.681343
14.77999	1.657089
14.89999	1.633378
15.01999	1.610192
15.13999	1.587516
15.25999	1.565334
15.49999	1.522397
15.61999	1.501614
15.73999	1.481271
15.85999	1.461354
15.97999	1.441853
16.09999	1.422754
16.21999	1.404048
16.33999	1.385723
16.45999	1.367768
16.57999	1.350174
16.69999	1.33293
16.81999	1.316028
16.93999	1.299459
17.05999	1.283212
17.17999	1.267281
17.3	1.251656
17.42	1.236329
17.54	1.221294
17.66	1.206542
17.78	1.192067
17.9	1.177861
18.02	1.163917
18.14	1.150229
18.26	1.136791
18.38	1.123597
18.5	1.11064
18.62	1.097914
18.74001	1.085415
18.86001	1.073137
18.98001	1.061074
19.10001	1.049221

X	Y
19.22001	1.037574
19.34001	1.026127
19.46001	1.014876
19.58001	1.003817
19.70001	.9929448
19.82001	.9822552
19.94001	.9717441
20.06001	.9614076
20.18002	.9512417
20.30002	.9412426
20.42002	.9314066
20.54002	.9217302
20.66002	.9122097
20.78002	.9028419
20.90002	.8936235
21.02002	.884551
21.14002	.8756217
21.26002	.8668321
21.38002	.8581796
21.50002	.8496612
21.62003	.841274
21.74003	.8330153
21.86003	.8248825
21.98003	.816873
22.10003	.8089842
22.22003	.8012137
22.34003	.7935591
22.46003	.7860181
22.58003	.7785883
22.70003	.7712676
22.94003	.7569447
23.06004	.7499384
23.18004	.7430328
23.30004	.7362259
23.42004	.7295159
23.54004	.7229009
23.66004	.716379
23.78004	.7099485
23.90004	.7036077
24.02004	.6973548
24.14004	.6911883
24.26004	.6851064
24.38004	.6791077

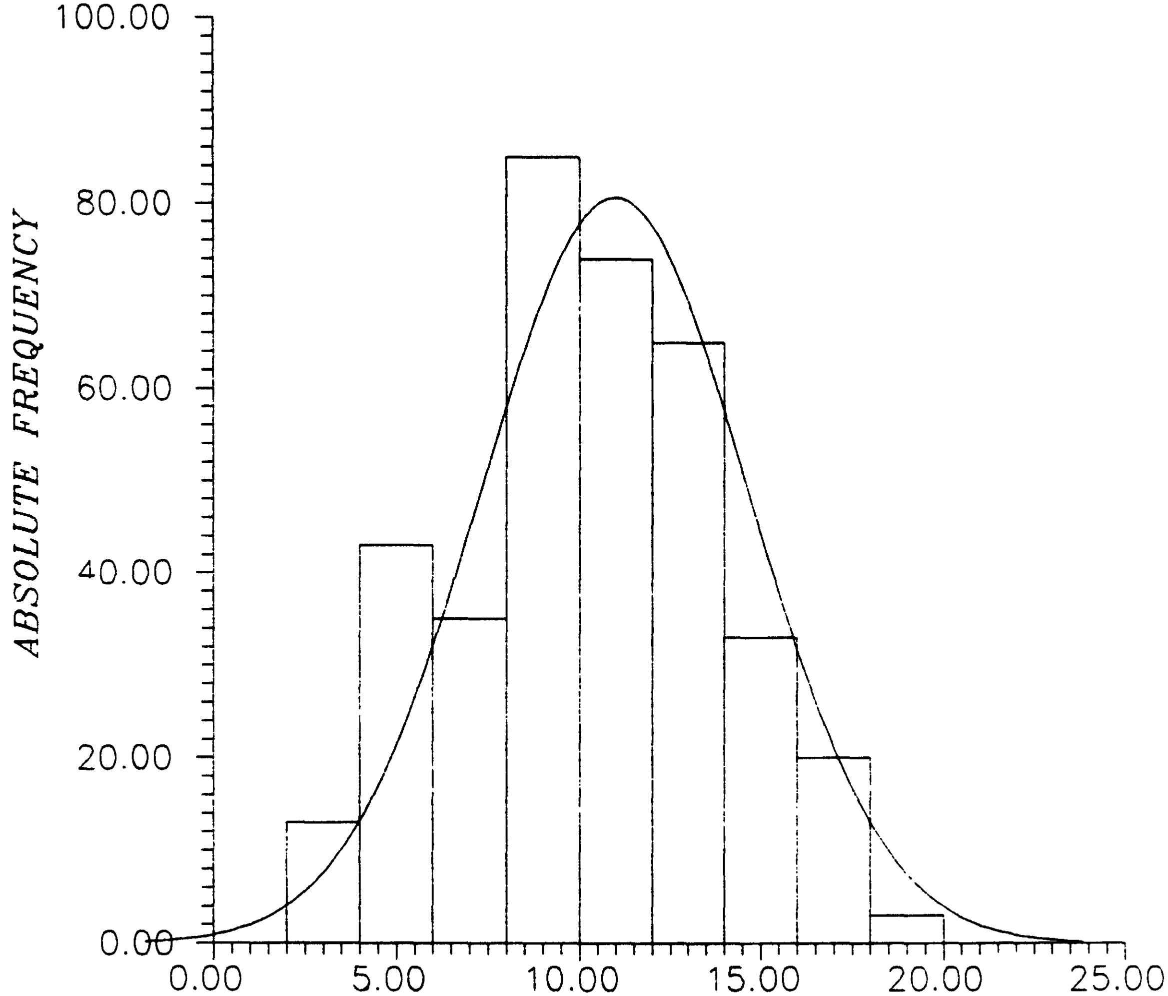


FIGURE E1.--Example plot of histogram and normalized curve overlay

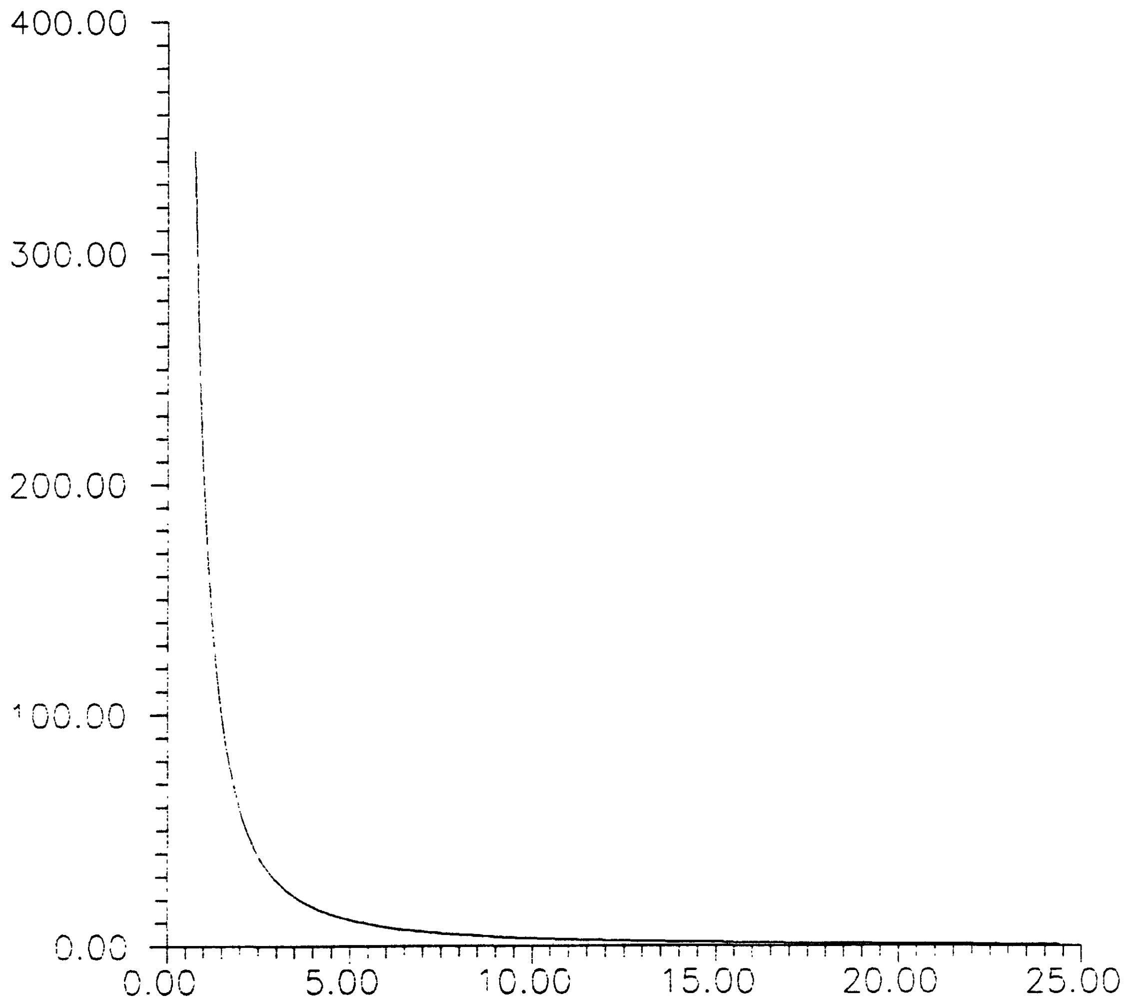


FIGURE E2.--Plot of FREQFIT generated best-fit curve output file (EX2POW.DAT)

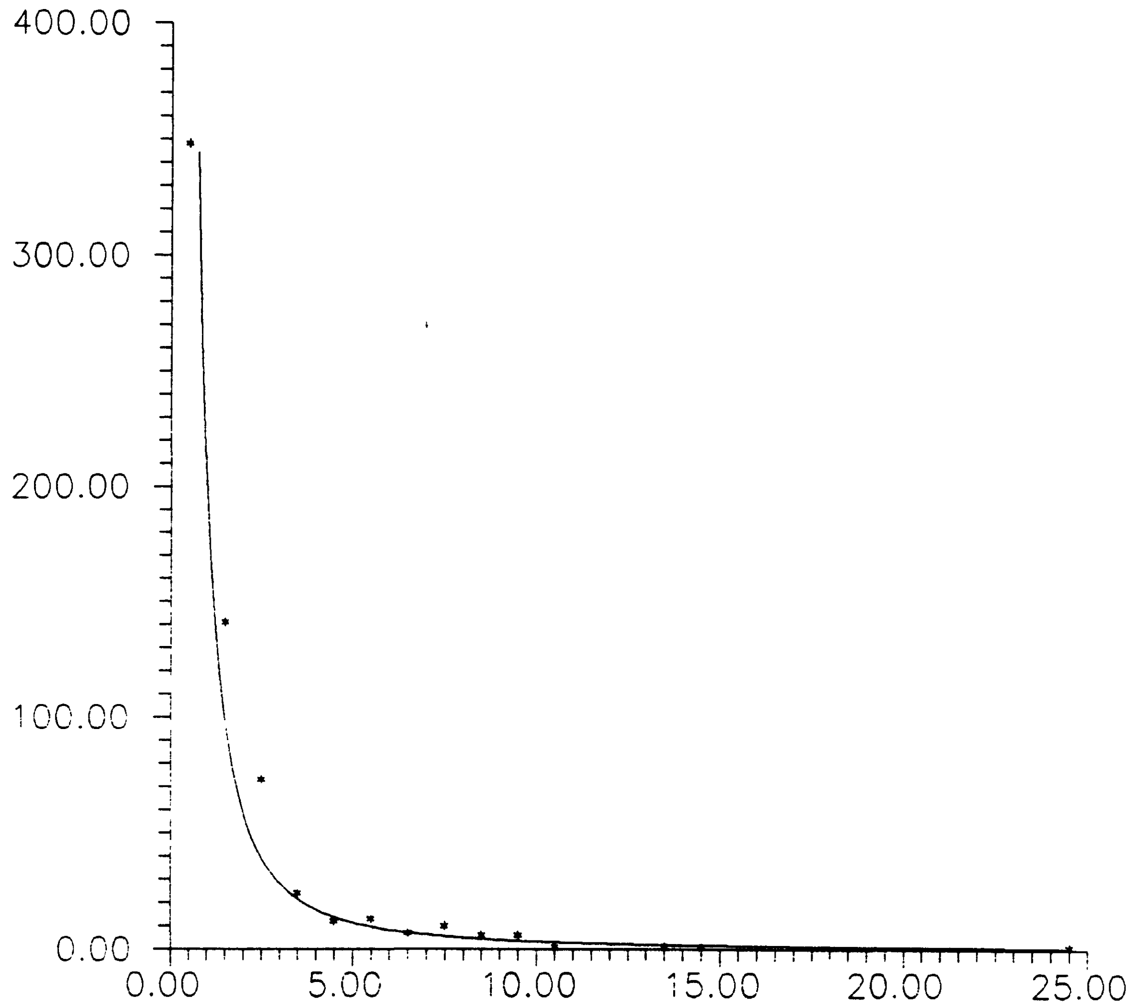


FIGURE E3--Plot of EXAMPLE2.DAT with best-fit curve overlay (EX2POW.DAT)

APPENDIX F--Equations used for statistical and regression analysis

HISTOGRAM (Hewlett-Packard, 1982):

Histogram construction and analysis considerations:

1. Missing values are not considered to be significant in any calculation, and are not considered when constructing histogram cells.

2. At least four cells are required to perform the chi-squared goodness of fit test.

3. If the number of cells desired for the histogram is unknown, the program will default this value to:

int[1 + 3.3 log10(number of valid measurements)]

STATISTICAL MEASUREMENTS (Hewlett-Packard, 1982):

Let:

X_i = ith measurement of the original one-dimensional data set

N = number of valid measurements

MEAN (Hewlett-Packard, 1982):

$$\overline{X} = \sum_{i=1}^{N} X_i/N$$

STANDARD DEVIATION (Hewlett-Packard, 1982):

$$Sdev = \sqrt{\frac{\sum_{i=1}^{N}(X_i - \overline{X})^2}{N-1}}$$

NORMAL CURVE OVERLAY (Hewlett-Packard, 1982):

$$NCO = \frac{100 * (\text{cell width}) * (EXP((X_i-\overline{X})^2 \;/\; (-2 * Sdev^2))}{\sqrt{2 * \Pi * Sdev}}$$

CHI-SQUARED (Hewlett-Packard, 1982):

$$X^2\ df = \sum_{i=1}^{N} \frac{(OFC_i - EFC_i)^2}{EFC_i}$$

where: df is the degrees of freedom = (# of cells) - 3 (1 degree of freedom is lost for number of cells, 1 for the estimated mean, and one for the estimated variance).

OFC_i is the Observed Frequency of Cell i
(the number of measurements which fall in cell i)

EFC_i is the Expected frequency of cell i
(the area under the normal curve overlay which would fall in cell i):

$$EFC_i = \int_{(i=1) * \text{cell width}}^{i * \text{cell width}} NCO(i)\ dw$$

PROBABILITY (Hewlett-Packard, 1982):

$$P(X^2 v > \text{calculated value}) = \frac{1 - \left(\frac{X^2}{2}\right)^{v/2} \exp\left(\frac{-X^2}{2}\right)}{\Gamma\left(\frac{v+2}{2}\right)} \left[1 + \sum_{j=1}^{\infty} \frac{X^{2j}}{(v+2)(v+4)\ldots(v+2j)}\right]$$

where: v = number of degrees of freedom
Γ = the standard gamma function
$\Gamma(.5) = .886226925$

The sum is calculated until the percentage of change between two consecutive sums is less than .000001 or $j = 30$.

REGRESSION EQUATIONS (Draper, 1959; Ezekiel, 1959): N = number of valid X,Y points

LINEAR REGRESSION: Y = b X + a

$$b = \left[\sum_{i=1}^{N} x_i y_i - \frac{\sum_{i=1}^{N} x_i \sum_{i=1}^{N} y_i}{N}\right] / \left[\sum_{i=1}^{N} (x_i)^2 - \frac{\left(\sum_{i=1}^{N} (x_i)\right)^2}{N}\right]$$

$$= \left[\sum_{i=1}^{N} x_i y_i - \frac{\sum_{i=1}^{N} x_i \sum_{i=1}^{N} y_i}{N}\right] / \left[\sum_{i=1}^{N} (x_i)^2 - N(\bar{X})^2\right]$$

$$a = \frac{\sum_{i=1}^{N} y_i}{N} - b * \frac{\sum_{i=1}^{N} x_i}{N}$$

$$= \bar{Y} - b\,\bar{X}$$

$$R = \frac{\left[\sum_{i=1}^{N} x_i y_i - \frac{\sum_{i=1}^{N} x_i \sum_{i=1}^{N} y_i}{N}\right]^2}{\left[\sum_{i=1}^{N} (x_i)^2 - \frac{\left(\sum_{i=1}^{N} (x_i)\right)^2}{N}\right]\left[\sum_{i=1}^{N} (y_i)^2 - \frac{\left(\sum_{i=1}^{N} (y_i)\right)^2}{N}\right]}$$

EXPONENTIAL REGRESSION: Y = a exp(bX)

$$b = \frac{\sum_{i=1}^{N} X_i \ln(Y_i) - \sum_{i=1}^{N} X_i \sum_{i=1}^{N} \ln(Y_i)/N}{\sum_{i=1}^{N} (X_i)^2 - \left(\sum_{i=1}^{N} X_i\right)^2/N}$$

$$a = \frac{\sum_{i=1}^{N} \ln(Y_i)}{N} - \frac{b\sum_{i=1}^{N} X_i \sum_{i=1}^{N} \ln(Y_i)}{N}$$

$$R = \frac{\left[\sum_{i=1}^{N} X_i \ln(Y_i) - \sum_{i=1}^{N} X_i \sum_{i=1}^{N} \ln(Y_i)/N\right]^2}{\left[\sum_{i=1}^{N} (X_i)^2 - \left(\sum_{i=1}^{N} X_i\right)^2/N\right]\left[\sum_{i=1}^{N} (\ln(Y_i))^2 - \left(\sum_{i=1}^{N} \ln(Y_i)\right)^2/N\right]}$$

POWER REGRESSION: $Y = a\,X^{b}$

$$b = \frac{\left[\sum_{i=1}^{N} \ln(X_i)\ln(Y_i) - \sum_{i=1}^{N} \ln(X_i) \sum_{i=1}^{N} \ln(Y_i)/N\right]}{\left[\sum_{i=1}^{N}\left(\ln(X_i)\right)^2 - \left(\sum_{i=1}^{N} \ln(X_i)\right)^2/N\right]}$$

$$a = \sum_{i=1}^{N} \ln(Y_i) - b * \sum_{i=1}^{N} \ln(X_i)/N$$

$$R = \frac{\left[\sum_{i=1}^{N} \ln(X_i)\ln(Y_i) - \sum_{i=1}^{N} \ln(X_i) \sum_{i=1}^{N} \ln(Y_i)/N\right]^2}{\left[\sum_{i=1}^{N} \ln(X_i)^2 - \left(\sum_{i=1}^{N} \ln(X_i)\right)^2/N\right]\left[\sum_{i=1}^{N}\left(\ln(Y_i)\right)^2 - \left(\sum_{i=1}^{N} \ln(Y_i)\right)^2/N\right]}$$

LOGARITHMIC REGRESSION: $Y = a + b \ln(X)$

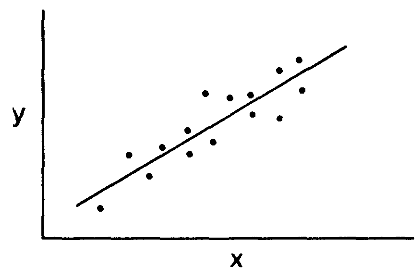
$$b = \frac{\left[\sum_{i=1}^{N} Y_i \ln(X_i) - \sum_{i=1}^{N} \ln(X_i) \sum_{i=1}^{N} Y_i\right]}{\sum_{i=1}^{N} \left(\ln(X_i)\right)^2 - \left(\sum_{i=1}^{N} \ln X_i\right)^2 /N}$$

$$a = \left[\sum_{i=1}^{N} Y_i - b * \sum_{i=1}^{N} \ln(X_i)/N\right]$$

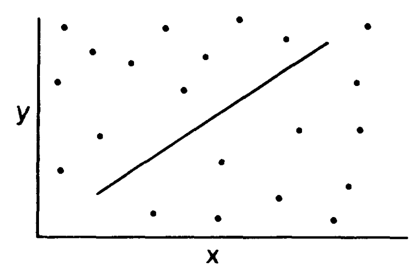
$$R = \frac{\left[\sum_{i=1}^{N} Y_i \ln(X_i) - \sum_{i=1}^{N} \ln(X_i) \sum_{i=1}^{N} Y_i /N\right]^2}{\left[\sum_{i=1}^{N} (\ln(X_i))^2 - \left(\sum_{i=1}^{N} \ln(X_i)\right)^2 /N\right]\left[\sum_{i=1}^{N} (Y_i)^2 - \left(\sum_{i=1}^{N} Y_i\right)^2 /N\right]}$$

CORRELATION COEFFICIENT (Freedman, 1979);

The CORRELATION coefficient, R, is a unitless number which measures how close data fit a given best-fit line. For example, consider the following diagram:

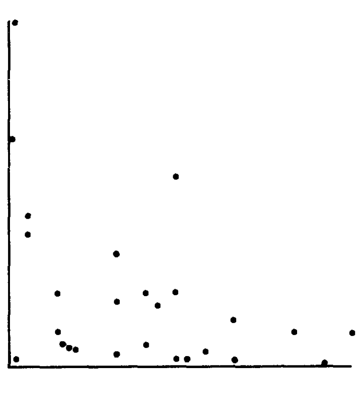


a) CORRELATION near one means data cluster tightly near the line

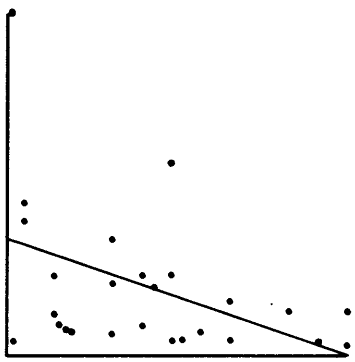


b) CORRELATION near zero means data do not cluster tightly near the line

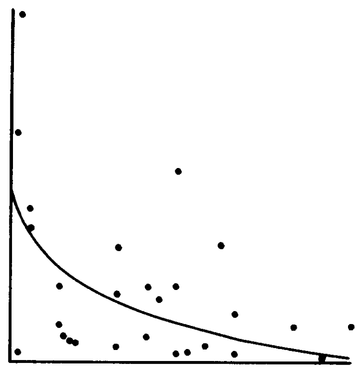
The CORRELATION coefficient should be used in conjunction with visual inspection to determine which regression line best fits a given data set. Because the distribution of *y* values influences the fitting of regression curves (Hoaglin and Welsch, 1978), comparison between correlation coefficients alone, to determine which regression line best fits a given data set, is not advised.



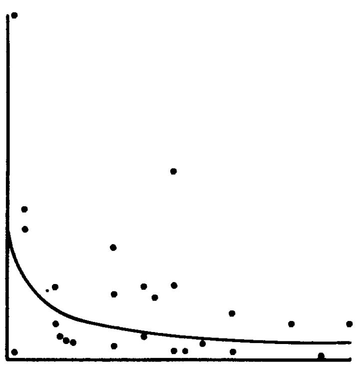
a) Original data set



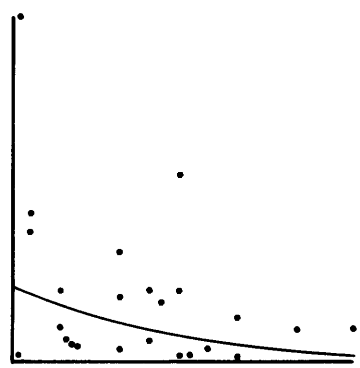
b) Linear regression
R = 0.21



c) Logarithmic regression
R = 0.34



d) Power regression
R = 0.26



e) Exponential regression
R = 0.25

From this example, it is easy to see by visual inspection that the logarithmic regression provides a line which best fits this data; note that it's correlation coefficient also has the highest value.

The correllation coefficient is given by:

$$R = \frac{\sum_{i=1}^{N} \frac{(X_i - \overline{X}}{Sdev\ x} * \frac{Y_i - \overline{Y})}{Sdev\ y}}{N}$$

☆U.S. GOVERNMENT PRINTING OFFICE: 1990 - 776-313/25147