A Curve-Fitting Program Using a Focal Interpreter on the KIM-1

BY THEODORE E. BRIDGE
54 Williamsburg Dr.
Springfield, MA 01108

I have three reasons for writing this article. First I want to report on my experience with the FCL65E interpreter that I purchased from the 6502 Program Exchange, 2920 Moana, Reno, NV 89509:

Program (HYPERTAPE on cassette)	$19.00
Mini Manual	6.00
User's Manual	12.00
Listing	35.00
Total	$72.00

Not being able to locate a suitable BASIC interpreter for my KIM-1, I decided to give this a try. My first project was this curve fitting program. I believe the FCL65E interpreter is more compact than BASIC, possibly a little slower, but quite adequate. It has excellent diagnostics. It even has string functions.

My second reason was that I would like to see a BASIC version of this program published in *DDJ*, hopefully with a timing comparison. With my program, it took 40 seconds to calculate the regression coefficients in example 1 with six degrees of freedom, and 16 seconds in example 2 with four degrees of freedom. I think there may be many readers who would like to have a BASIC version of this program.

My third reason is that I want to donate this program to the public domain. It is very handy if you want to read data from tables or curves, or even if you want to make sense out of a large mass of test data.

I once used a FORTRAN version of this program to analyze a large block of test data on the performance of a very big centrifugal pump. None of us could make any sense out of the data, but the computer could. It showed that the efficiency peaked at the end of the first month, and then started to fall. We were then able to optimize the maintenance schedule on the pump. You could use a similar program to optimize the spark and throttle setting on a gas engine as a function of speed, load, and temperature.

In the listing, I present a FOCAL program for finding a formula that will relate a dependent variable, W, to two independent variables, X and Y. When we say $W = f(X, Y)$ we simply mean that the variable W depends in a definite way on variables X and Y. For any value of X and Y there is one and only one possible value to W. To make this analysis, we assume that a polynomial equation can be found to express the relationship.

We will now consider an example having 3 degrees of freedom (a parabolic fit) with respect to X, and 2 degrees of freedom (a linear fit) with respect to Y.

The polynomial equation will look like this: W will be the given value, and V the calculated value of the dependent variable:

$$V = C(0) + C(1)X + C(2)X^2 + (C(3) + C(4)X + C(5)X^2)\,Y$$

So you can see, if we knew the coefficients in the array, C, we could calculate a value of V for any combination of values for X and Y.

Our first task is to find values for these coefficients that will give a minimum error. Since there are 6 unknowns in the array, C, we will need 6 equations. These regression equations have been developed in the science of statistics and are not hard to program. We will express these equations in matrix form. For each data point (one value of X, Y, and W) we will develop a correction to be added to the matrix and will collect the total in array, T.

We will build the first row of the correction matrix in array, U, and each of the following terms will represent consecutive elements in this array. There are 7 terms representing this first equation. The last term is the given value of the dependent varaible, W. Thus:

$$U = 1,\ X,\ X^2,\ Y,\ YX,\ YX^2,\ W$$

Using the following algorithm, we will build the regression matrix in array, T. It will have 6 rows and 7 columns. We initialized array T = 0. For each data point, we will update every element of array T as follows:

$$T(I,J) = T(I,J) + U(I) * U(J)$$

I varies from zero to M1, and J varies from zero to M. Let X1 = the degrees of freedom for X, and Y1 = the degrees of freedom for Y. Then M = X1 * Y1, and M1 = M − 1.

We must have at least M unique data points—as many as we have total degrees of freedom. That is, at least one data point for every row in the matrix T. The more, the better. After all data points have been processed, we will perform a Gaussian elimination to solve the simultaneous equations for the regression coefficients.

This program was written in FCL65E (FOCAL) for the KIM-1. FOCAL is very similar to BASIC:

D for (DO)	in FCL65	is like	GOSUB	in BASIC
C " (COMMENT)	"	" "	REM	"
F " (FOR)	"	" "	FOR	"
G " (GO)	"	" "	GOTO	"
I " (IF)	"	" "	IF	"
S " (SET)	"	" "	LET	"
T " (TYPE)	"	" "	PRINT	"
A " (ASK)	"	" "	INPUT	"

The DO statement in FOCAL assumes an implied return at the end of the line (i.e., D 6.8 in line 5.7 of the listing); or, at the end of the group called (i.e. D 3 at the end of line 7.1). However, a return will be made whenever the command R is encountered. The FOR statement in FCL65E assumes an implied NEXT command at the end of the line.

The IF statement in FCL65E is like a FORTRAN II IF in that it cannot handle logical operators. Rather, the expression within the parentheses is evaluated. A branch is made to one of three line numbers that follow the closing parenthesis, depending on the value of the expression. It branches to the first line number if the expression is negative, to the second if it is zero, and to the third if it is positive. If any of these line numbers are missing, control drops through to the next instruction.

I believe that the comments on the listing and on the examples that follow will complete the explanation of the program and its use.

Listing

```
C FOCAL-65 (V3D) 26-AUG-77

1.10 S X1=3,Y1=2,P=2.3,Q=1;C HERE,WE SPECIFY THE DEGREES
1.11 C  OF FREEDOM FOR EACH IND. VARIABLE
1.14 T %1,!!!,"EXAMPLE NO. ",Q,!!
1.20 S L=5,M=X1*Y1,M1=M-1,M2=M↑2,X2=X1-1,Y2=Y1-1;G 10.1

2.01 C SQUARE ROOT ROUTINE
2.10 S &3=.000001;I (&-&3↑2)2.3;S &1=&,&2=2
2.20 S &2=&1/&;I (FABS(&2-&)-&*&3)2.3;S &=(&+&2)/2;G 2.2
2.30 R

3.09 C GAUSSIAN ELIMINATION ROUTINE
3.10 I (K-I)3.2,3.3,3.2
3.20 S I5=I+J,T(I5)=T(I5)-T(K1)*T(I+K2)
3.30 R

4.01 C FUNCTION TO PRINT IN SCIENT. NOTATION
4.10 S &1=0;I (-&)4.2,4.9;T "-";S &=-&
4.20 I (&-1)4.7;I (&-10)4.9;S &1=&1+1,&=&/10;G 4.2
4.70 S &1=&1-1,&=&*10;G 4.2
4.90 T %1.05,&,"E",%1,&1," ";R

5.09 C IN U, WE BUILD FIRST ROW OF REGRESSION MATRIX
5.10 S B=1;F J=0,X1,M1;D 5.6;S B=B*Y
5.15 S U(M)=W
5.17 C IN T, WE BUILD THE MATRIX
5.19 F J=0,M;F I=0,M1;S I1=I+J*M,T(I1)=T(I1)+U(I)*U(J)
5.40 T !; R
5.60 S A=1;F I=0,X2;S U(I+J)=A*B,A=A*X
5.70 T !;F C=0,M,M2;D 6.8;COPY MATRIX

6.09 COLUMN C IS PRINTED IN LINE 6.8
6.40 S R=G+R1;I (M1-R)6.7
6.60 S Z=FSBR(4,T(R+C));COPY IN SCIENT NOTATION
6.70 R
6.80 T !;F R1=0,6,M1;T !;F G=0,5;D 6;COPY 6 NOS./LINE

7.01 COEFFICIENTS ARE PRINTED AFTER SOLVING EQUATIONS
7.10 F K=0,M1;D R;T 2;S K2=K*M,A=T(K+K2);F J1=K,M1;D 7.5;F I=0,M1;D 3
7.15 T !!!,"REGRESSION COEFFS."
7.20 T !;S C=M2;D 6.8
7.30 T !;R;C K1 POINTS TO ROW K,COL J; K2 TO ROW 0,COL K
7.50 S J=(J1+1)*M,K1=K+J,T(K1)=T(K1)/A

8.09 C REREAD DATA AND TABULATE RESULTS
8.10 S V=0,K=M1+M2;F J=1,Y1;S V=V*Y;D 8.4;S V=V+V1
8.20 S E=V-W,E2=E2+E↑2,N=N+1;T !,N,"  "
8.30 S A=FSBR(4,X),A=FSBR(4,Y),A=FSBR(4,W),A=FSBR(4,V),A=FSBR(4,E)
8.35 R
8.40 S V1=0;F I=1,X1;S V1=V1*X+T(K),K=K-1
8.50 I (L-8)8.7
8.60 T !!!,"RMS ERROR ";S A=FSBR(4,FSBR(2,E2/N));T !;G 20.1
8.70 D 7
8.80 T !!!,"          X          Y          W        CALC       ERROR"
8.90 S L=8,E2=0,N=0;G 10.1

10.01 COORDINATES FOR 6 DATA POINTS ARE ENTERED HERE. THEY ARE READ FROM
10.03 CURVE 1. THIS DATA IS READ TWICE BY THE PROGRAM:FIRST TO BUILD THE
10.05 C   6X6  REGRESSION MATRIX, AND AGAIN TO SOLVE FOR THE REGRESSION
10.07 COEFFICIENTS. FOR EACH POINT, THE CALCULATED VALUE V IS TO BE
10.09 COMPARED TO THE GIVEN VALUE W.
10.10 S X=1,W=3,Y=1;D L
10.20 S X=2,W=4;D L
10.25 S X=0,W=.75;D L
10.30 S X=1,W=6,Y=2;D L
10.40 S X=2,W=11;D L
10.60 S X=0,W=3.764;D L

20.01 G 8.5
20.02 C HERE, WE PLOT A GRAPH OF THE RESULTS
20.10 T !!!;F X=0,.1,2;T !;D 20.4;D 21
20.20 T !!!,"CURVE  ",Q,"  ---  W  VS  X,Y";T !!;Q
20.30 E ;S X1=2,Y1=2,P=5.7,Q=2;G 1.14
20.40 F Y=1,.5,2;D 8.1;S W(2*Y)=V*5+.5

21.04 S Z3=FINT(X+.05)-X
21.05 S Z=-1,A=32;I (Z3)21.1;S A=45
21.10 T "  ";F N=1,W(2);D 22
21.20 D 22.3;F N1=N+1,W(3);D 22
21.30 D 22.3;F N=N1+1,W(4);D 22
21.40 D 22.3

22.10 I (Z)22.2;S Z=Z-1,S1=FOUT(A);R
22.20 S Z=8;T "I";R
22.30 T ".";I (Z)22.4;S Z=Z-1;R
22.40 S Z=8;R
```

EXAMPLE NO. 1
The following magic number was typed by the program just to let you know that something is happening during the long wait.

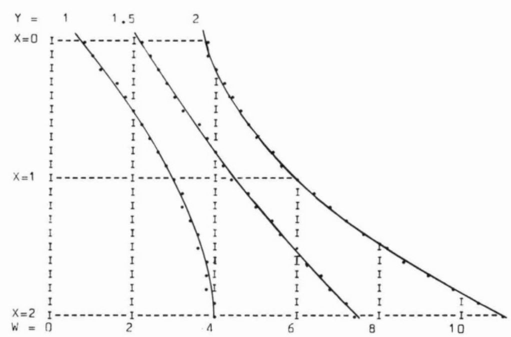
```
11111111222222
```

It took 40 seconds to calculate the
REGRESSION COEFFS.

```
-2.26400E0 4.89600E0 -2.63200E0 3.01400E0 -2.02100E0 2.00700E0
```

```
        X           Y           W          CALC       ERROR
1  1.00000E0 1.00000E0 3.00000E0 3.00000E0 -8.94070E-8
2  2.00000E0 1.00000E0 4.00000E0 4.00000E0 2.23517E-8
3  0.00000E0 1.00000E0 7.50000E-1 7.50000E-1 -2.79397E-8
4  1.00000E0 2.00000E0 6.00000E0 6.00000E0 8.19564E-8
5  2.00000E0 2.00000E0 1.10000E1 1.10000E1 -4.47035E-8
6  0.00000E0 2.00000E0 3.76400E0 3.76400E0 -7.45058E-9
```

RMS ERROR 0.000000E0



EXAMPLE NO. 2 In this example, we use the same data, but with 2 (not 3) degrees of freedom for variable X.

```
111111

6.00000E0 6.00000E0 9.00000E0 9.00000E0

6.00000E0 1.00000E1 9.00000E0 1.50000E1

9.00000E0 9.00000E0 1.50000E1 1.50000E1

9.00000E0 1.50000E1 1.50000E1 2.50000E1

2.85140E1 3.90000E1 4.92780E1 6.70000E1 2
```

You may find this printout useful for debugging, should you wish to translate this program into BASIC for use on your machine. It is printed before each sweep while solving the equations for the regression coefficients in group seven lines of the program.

```
6.00000E0 6.00000E0 9.00000E0 9.00000E0

1.00000E0 4.00000E0 0.00000E0 6.00000E0

1.50000E0 0.00000E0 1.50000E0 1.50000E0

1.50000E0 6.00000E0 1.50000E0 1.15000E1

4.75233E0 1.04860E1 6.50700E0 2.42290E1 2

6.00000E0 6.00000E0 9.00000E0 9.00000E0

1.00000E0 4.00000E0 0.00000E0 6.00000E0

1.50000E0 0.00000E0 1.50000E0 1.50000E0

0.00000E0 1.50000E0 1.50000E0 2.50000E0

2.13083E0 2.62150E0 6.50700E0 8.50000E0 2

6.00000E0 6.00000E0 9.00000E0 9.00000E0

1.00000E0 4.00000E0 0.00000E0 6.00000E0

1.50000E0 0.00000E0 1.50000E0 1.50000E0

-1.50000E0 1.50000E0 1.00000E0 1.00000E0

-4.37617E0 2.62150E0 4.33800E0 1.99300E0 2
```

It took 16 seconds to calculate the
REGRESSION COEFFS.

```
-1.38667E0 -3.68000E-1 2.34500E0 1.99300E0
```

```
        X           Y           W          CALC       ERROR
1   1.00000E0 1.00000E0 3.00000E0 2.58333E0 -4.16667E-1
2  2.00000E0 1.00000E0 4.00000E0 4.20833E0 2.08333E-1
3  0.00000E0 1.00000E0 7.50000E-1 9.58333E-1 2.08333E-1
4  1.00000E0  2.00000E0 6.00000E0 6.92133E0 9.21333E-1
5  2.00000E0 2.00000E0 1.10000E1 1.05393E1 -4.60667E-1
6  0.00000E0 2.00000E0 3.76400E0 3.30333E0 -4.60667E-1
```

RMS ERROR 5.05585E-1 Since we are trying to fit a straight line to parabolic data, we would expect some error.