Thermochimica Acta, 75 (1984) 183–196
Elsevier Science Publishers B.V., Amsterdam – Printed in The Netherlands

A WEIGHTED LEAST SQUARES COMPUTATION OF THE ANTOINE CONSTANTS FROM VAPOR PRESSURE DATA USING A MICROCOMPUTER

JAMES E. LUNDEEN and J.E. HOUSE, JR. *

Department of Chemistry, Illinois State University, Normal, IL 61761 (U.S.A.)

(Received 16 November 1983)

ABSTRACT

The purpose of this work was to develop an accurate, rapid method for determining the constants A, B, and C of the Antoine equation, $\log p = A - B/(C+t)$ which relates the vapor pressure of a liquid to temperature. The procedure makes use of a weighted least squares curve fit with an iterative computation. The computation has been programmed in BASIC for use on a microcomputer, and the details of the program and its use are presented. A comparison of the results of the present method with published values for several compounds indicates that the constants are determined more accurately by the present method than by other methods.

INTRODUCTION

Some of the most fundamental thermodynamic properties of liquids are obtained from vapor pressure data, and the most reliable equation for representing vapor pressure as a function of temperature is the Antoine equation [1,2]. That equation can be written as

$$\log p = A - \frac{B}{C+t} \qquad (1)$$

where A, B, and C are constants characteristic of the liquid, p is the vapor pressure (Torr), and t is the temperature (°C). Compilations of A, B, and C values are available [3–5] and these constants are used in the computation of other properties, such as the cohesion energy and heat of vaporization of the liquid [6,7]. Therefore, the Antoine equation is extremely useful in describing a variety of thermodynamic properties.

Typically, the vapor pressure of a liquid is measured at different temperatures, or the boiling point is measured at different pressures. From these data, the constants A, B, and C must be determined. Many of the tabulated

---

* Author for correspondence.

0040-6031/84/$03.00

values have been determined by means of a graphical method which is based on the determination of the boiling points at three different pressures [1]. When employing such a method, the data must be very accurate, or sizeable errors will result in the calculated Antoine constants. It is preferable to employ a least squares best fit procedure in order to minimize errors in the calculated constants. It is also important to use many more than three $[p, t]$ data pairs. Large compilations of vapor pressure data are readily available for many compounds [8,9]. However, a least squares best fit is difficult to obtain, because the Antoine equation is not linear and non-linear least squares procedures are not widely available for use on a programmable calculator or microcomputer. This is unfortunate, since programmable calculators and microcomputers are widely available and have sufficient calculation power to handle the computation.

Trump [2] has shown the superiority of the Antoine equation over the 3-term Chebyshev polynomial equation. Trump's program was based on the method of Willingham et al. [10]. We show the superiority of the present method to that of Willingham et al. [10]. The method presented here is easily programmed in BASIC for use on a microcomputer and is also easily programmed on a pocket-sized programmable calculator. In this paper, we present the details of a computational method for determining the Antoine constants by a weighted non-linear least squares best fit and a program written in BASIC. The present computational method can utilize a large number of $[p, t]$ data pairs. As previously mentioned, several physical properties have a dependence on temperature given by the form of eqn. (1). Therefore, the present method is a powerful curve fitting procedure for several physical properties.

PRINCIPLES

If we let $Y_i = \log(p_i)$ and $\beta_i = 1/(C + t_i)$, then eqn. (1) becomes

$$Y_i = A - B\beta_i \tag{2}$$

Let Q be the square of the difference between the Y_i value and the right hand side of eqn. (2), then

$$Q_i = w_i(Y_i - A + B\beta_i)^2 \tag{3}$$

where $w_i = w_i'[Y_i \ln(10)]^2$ and $w_i' = 1$ for all i (for equal weighting of data). Alternatively, w_i' could be an integer value greater than one, if replicate pressure values were determined at certain temperatures. Yet another choice for w_i' could be 1/(variance in pressure), if three or more replicate pressure values exist at certain temperatures. The choice of weighting factors, w_i', is a matter of preference. Equation (3) represents the sum of squares of differences between the best fit curve through the $[p, t]$ data pairs and the actual data points. The weighting factors will be described further later in this paper.

Expansion of eqn. (3) and representation in terms of the sums of individual errors gives

$$Q = \Sigma w_i Y_i^2 - 2A\ \Sigma w_i Y_i + 2B\ \Sigma w_i Y_i \beta_i - 2AB\ \Sigma w_i \beta_i + A^2\ \Sigma w_i + B^2 w_i \beta_i^2 \qquad (4)$$

In order for the errors to be minimized, we derive the expression for the partial derivative of Q with respect to A and set the resulting expression equal to zero

$$\frac{\partial Q}{\partial A} = 0 = -\Sigma w_i Y_i - B\ \Sigma w_i \beta_i + A\ \Sigma w_i \qquad (5)$$

All summations are from 1 to N, where N is the number of $[p, t]$ data pairs. Solving eqn. (5) for A yields

$$A = \frac{\Sigma w_i Y_i + B\ \Sigma w_i \beta_i}{\Sigma w_i} \qquad (6)$$

An expression for the partial derivative of Q with respect to B is obtained in a similar fashion

$$\frac{\partial Q}{\partial B} = 0 = \Sigma w_i Y_i \beta_i - A\ \Sigma w_i \beta_i + B\ \Sigma w_i \beta_i^2 \qquad (7)$$

Solving eqn. (7) for B yields

$$B = \frac{A\ \Sigma w_i \beta_i - \Sigma w_i Y_i \beta_i}{\Sigma w_i \beta_i^2} \qquad (8)$$

Each of the variables, A (eqn. 6) and B (eqn. 8), are in terms of the other and also in terms of C. Therefore, eqns. (6) and (8) cannot be used directly to compute the constants A, B, and C. An expression for the partial derivative of Q with respect to C is equated with zero

$$\frac{\partial Q}{\partial C} = 0 = \frac{\partial}{\partial C}\left(2B\ \Sigma w_i Y_i \beta_i - 2AB\ \Sigma w_i \beta_i + B\ \Sigma w_i \beta_i^2\right) \qquad (9)$$

which yields

$$0 = -\Sigma w_i Y_i \beta_i^2 + A\ \Sigma w_i \beta_i^2 - B\ \Sigma w_i \beta_i^3 \qquad (10)$$

Equations (5), (7), and (10) are the necessary equations for the computation of the Antoine constants from the $[p, t]$ data.

Solving eqn. (10) for B yields

$$B = \frac{-\Sigma w_i Y_i \beta_i^2 + A\ \Sigma w_i \beta_i^2}{\Sigma w_i \beta_i^3} \qquad (11)$$

Equating the right hand side of eqns. (8) and (11) yields

$$\frac{A\ \Sigma w_i \beta_i^2 - \Sigma w_i Y_i \beta_i^2}{\Sigma w_i \beta_i^3} = \frac{A\ \Sigma w_i \beta_i - \Sigma w_i Y_i \beta_i}{\Sigma w_i \beta_i^2} \qquad (12)$$

Solving eqn. (12) for A yields

$$A = \frac{\left(\Sigma w_i \beta_i \, \Sigma w_i Y_i \beta_i^2\right) - \left(\Sigma w_i Y_i \beta_i \, \Sigma w_i \beta_i^3\right)}{\left(\Sigma w_i \beta_i^2 \, \Sigma w_i \beta_i^2\right) - \left(\Sigma w_i \beta_i \, \Sigma w_i \beta_i^3\right)} \tag{13}$$

Employing eqn. (13), the least squares best fit value of A may be computed for any value of C.

In a similar manner, solving eqn. (10) for A and equating the resulting expression with the right hand side of eqn. (7) yields

$$B = \frac{\left(\Sigma w_i \, \Sigma w_i Y_i \beta_i^2\right) - \left(\Sigma w_i Y_i \, \Sigma w_i \beta_i^2\right)}{\left(\Sigma w_i \beta_i \, \Sigma w_i \beta_i^2\right) - \left(\Sigma w_i \, \Sigma w_i \beta_i^3\right)} \tag{14}$$

We now have two equations to compute A and B which are functions of the value of C (in terms of β_i) and the $[p, t]$ data only. These equations (13 and 14) are used in the iterative procedure. The relationships in eqns. (13) and (14), between A, B, and C, are not obvious from the method of Willingham et al. [10].

In the present method, a value is assumed for C and the values of A and B are calculated from eqns. (13) and (14), respectively. The iterative procedure employs eqn. (6) in the form

$$A - \frac{\Sigma w_i Y_i + B \, \Sigma w_i \beta_i}{\Sigma w_i} = 0 \tag{15}$$

The left hand side of eqn. (15) will equal zero only when the value of C is exactly the best fit value. Since this is an iterative procedure, the value of C will never become exactly equal to the best fit value. Therefore, eqn. (15) is used in the form

$$A - \frac{\Sigma w_i Y_i + B \, \Sigma w_i \beta_i}{\Sigma w_i} = \delta_j \tag{16}$$

where j is the iteration (0, 1, 2,...), δ_j is thus a measure of the accuracy of C. However, neither the sign nor magnitude of δ_j is used directly as a criterion of convergence.

CONVERGENCE

Previous least squares methods for determining the Antoine constants involve one or more undesirable features. The method of Willingham et al. [10] requires initial estimates of A, B and C, and correction factors are then computed to increment the constants to approximately their optimum values. This method also utilizes weighting factors which are calculated from average A and B values for many compounds; this is not desirable.

Another commonly used method of determining A, B, and C values utilizes matrices and also requires initial values of A, B, and C to begin its calculation [6,10]. This method yields matrices of variance and covariance values, and is applicable to solving many types of equations, especially non-linear equations. This method provides standard errors for the computed values. However, it suffers because the accuracy of its results depends on highly accurate initial estimations of A, B, and C, which may be difficult to estimate accurately.

The present method is a weighted non-linear least squares best fit method that does not use initial values of A or B, does not utilize matrices, does not require correction factors, and does not compute standard error values. Least squares best fit values for A and B are calculated exactly for any value of C. The initial value of C is estimated systematically and may be a very crude estimation. Since most liquids have C values in the range of 200–250, it is easy to select an initial C value which will be of the order of magnitude of the best fit value (final value). We have determined that a reasonable initial value of C is 200 for most compounds, and that a reasonable increment value for C, ΔC, is 100.

One of the equations used in the convergence procedure, eqn. (16), has negative values for C values less than the least squares best fit C value, and positive values for C values greater than the least squares best fit C value. Should C ever become exactly the best fit value, eqn. (16) will equal zero.

The procedure for determining the correct value of C involves making an initial estimate of C (usually 200), C_0, and then computing the value of δ_0. If the initial estimate of C is below the final best fit value, then δ_0 will have a negative value. However, if δ_0 has a positive value, then the initial estimate of C was too high and the procedure halves the initial estimate of C and restarts the computation. Provided that δ_0 was a negative number, we increment the trial value of C and compute the corresponding δ value. If the δ value is still negative, the trial value of C is once again increased by adding ΔC to the previous value. When the δ value finally becomes positive, then the estimate of C is greater than the best fit value. Therefore, one has both an upper and a lower boundary on the best fit value of C; restated, an interval, whose upper and lower limits are known, is known to contain the best fit value of C.

We next assume that the interval is sufficiently small that the δ values, which correspond to the C values in the interval, form a straight line. If they actually form a straight line, then the next iteration will yield the best fit value of C. However, in practice, the δ values, the error values that correspond to the C values, fit some curvilinear function within the interval. Therefore, several iterations will be necessary to successively reduce the size of the interval known to contain the best fit C value until the error values form a straight line through the best fit C value. We assume that the best fit value of C lies proportionally between the upper and lower limits of the interval as determined by the δ_j values. Thus if δ_{j-1} is the lower limit of the interval and the upper limit is δ_j, then next approximation for C, C_{j+1}, is computed from eqn. (17)

$$C_{j+1} = C_j - \{[\delta_j/(\delta_j - \delta_{j-1})](C_j - C_{j-1})\} \tag{17}$$

The method converges when the value of the constant A changes by less than a pre-determined amount. We have determined that 1.0×10^{-5} is a reasonable accuracy limit for A. However, this may be altered to suit the accuracy requirements, or to speed up the calculations. Therefore, when the value of A changes by less than the absolute value of 1.0×10^{-5} from one iteration to the next, the method has converged. However, if the change in the magnitude of A is greater than 1.0×10^{-5}, then either the lower or upper limit of the interval is re-defined. In particular, if δ_{j+1} is a negative number, then the lower limit on C, C_{j-1}, is set equal to C_{j+1}. Conversely, if the δ_{j+1} is a positive number, then the upper limit on C, C_j, is set equal to C_{j+1}. In either case, C_{j+2} and δ_{j+2} are calculated using eqn. (17) and the re-defined limits on the interval. C_{j+2} will become the new value for C_{j+1}, and δ_{j+2} the new value for δ_{j+1} in eqn. (17), respectively. This method is about five times faster than halving the interval in search of the best fit C value.

The method in this case is applied to an equation that is logarithmic in p. Therefore, the resulting sum of squared errors (SSE) in p, rather than log(p), will not yield the least squares result. Deming [11] described a procedure that corrects for the logarithmic p values by using weighting factors which are functions of the p values. We have employed Deming's procedure in this method. When fitting a least squares solution in log(p), weighting factors equal to $[p \ln(10)]^2$ will result in an equal weighting of data and yield a true least squares result [11]. Without Deming's procedure, the present method yields SSE values which are not consistently lower than other methods [6,10]. However, using Deming's procedure, the present method gives a consistently smaller SSE value than any other method, for any given data set. While it is true that other methods could yield results as accurate as the present method, such methods require extremely accurate initial values of all Antoine constants to be competitive, and in practice, the present method is consistently superior.

While the present method does not compute standard error values of the Antoine constants, it could be used to determine highly accurate constant values and then the same values used as initial estimates for another method which does compute the standard error values. The fact that the present method uses only a value of C as input with the $[p, t]$ data, and that it makes use of weighted least squares, makes it the simplest and most accurate method available for determining Antoine constants.

THE PROGRAM

The program to implement the present method (algorithm) was written in BASIC for use on a microcomputer. The program has been run on a TRS-80 Model I microcomputer and minor modifications may be required for the program to be run on other brands of microcomputers. In order to minimize roundoff errors, the computation is performed in double precision. This is especially important for the logarithmic values of p. The program inputs data from pre-established data files by "merging" such files with the main program prior to execution. The program listing is thoroughly commented and will not be described further.

The format for establishing the data files is as follows:

(1) the first line number must be larger than the last line number of the program;

(2) the first line contains the number of data points, N, to be input by the program;

(3) the next N lines contain one temperature value, per line, and its corresponding pressure and weighting values;

(4) the next line contains the initial estimate of C, the value for incrementing C, the reported A value, the reported B value, and finally, the reported C value;

(5) the last line contains a string variable which is the number of the compound.

A complete listing of the program which has been merged with a sample data set is given in the Appendix.

The results of the data analysis are printed as follows. The name of the compound is printed first. Next, labels for six columns of numbers are printed, followed by N rows of six numbers. Column one, labeled "T(I)", contains the values of temperature (°C), which were contained in the original data set. Column two, labeled "P(I)", contains the pressure values (Torr), which correspond to the temperature values in column one. Column three contains the values of pressure predicted by the present method for the corresponding temperature values in column one. Column four contains the differences between columns two and three, and represents the residual errors to be squared and summed (SUM OF SQUARES LSM). Column five contains the values of pressure predicted by the reported values of the Antoine constants. Column six contains the differences between columns four and five, and represents the residual errors to be squared and summed (SUM OF SQUARED REP).

Next, the boiling point of the compound at 760 Torr, based on the results of the present method, is printed; this is followed by the boiling point based on the reported values of the Antoine constants. The values of A, B, and C, which result from the present method, and then the reported values of A, B, and *C*, are printed. Finally, the sum of squared errors for the present method and that for the reported constants are printed.

RESULTS AND DISCUSSION

The method described here has been tested with the sets of data reported by Willingham et al. [10], and by Meyer et al. [6,7]. These data represent the results of high precision vapor pressure measurements and provide a critical test of data fitted to the Antoine equation. Table 1 shows the results obtained for data reported for *m*-xylene [10]. The results shown indicate that the present method is in excellent agreement with that of Willingham et al. [10]. Although six digits are shown in each of the constants reported by Willingham et al. [10], the full number of digits given in the original report [10] were used in computing SUM OF SQUARES REP. Table 1 shows that a smaller SSE was obtained using the constant values calculated by the present method. Similar results have been found with other data, and Table 2 shows a summary of the results obtained. In every case, the SSE from the

TABLE 1

Results obtained for *m*-xylene using published data (ref. 10)

T(I)	P(I)	P-LSM	DEL-P	P-REP	DEL-P REP
140.041	779.360	779.338	−0.022	779.346	−0.014
139.493	767.950	768.004	0.054	768.011	0.061
138.869	755.250	755.256	0.006	755.262	0.012
138.314	744.060	744.059	−0.001	744.064	0.004
137.713	732.090	732.082	−0.008	732.086	−0.004
132.128	627.930	627.910	−0.020	627.908	−0.022
124.205	500.700	500.677	−0.023	500.670	−0.030
116.896	402.440	402.411	−0.029	402.402	−0.038
110.041	324.940	324.931	−0.009	324.922	−0.018
103.396	261.750	261.778	0.028	261.772	0.022
97.870	217.170	217.190	0.020	217.185	0.015
91.860	175.890	175.923	0.033	175.921	0.031
87.367	149.440	149.460	0.020	149.459	0.019
82.522	124.670	124.676	0.006	124.676	0.006
77.747	103.650	103.662	0.012	103.664	0.014
73.558	87.740	87.722	−0.018	87.724	−0.016
70.458	77.280	77.281	0.001	77.284	0.004
67.123	67.230	67.223	−0.007	67.226	−0.004
63.436	57.410	57.395	−0.015	57.399	−0.011
59.203	47.670	47.620	−0.050	47.624	−0.046

EST. B.P. LSM > 139.1022, EST. B.P. REP > 139.1019.
A COMPUTED > 7.00172, *B* COMPUTED > 1457.11, *C* COMPUTED > 214.488.
A REPORTED > 7.00343, *B* REPORTED > 1458.21, *C* REPORTED > 214.609.
SUM OF SQUARES LSM = 0.1139E-01, SUM OF SQUARES REP = 0.1194E-01.

TABLE 2

Comparison of calculated constants with those reported for several compounds

Compound	Present method				Reported values[a]			
	A	*B*	*C*	SSE	*A*	*B*	*C*	SSE
Butyronitrile	6.26054	1456.33	224.658	2.85767E-04	6.25397	1452.08	224.186	3.86163E-04
Benzene	6.89116	1202.67	219.790	8.05942E-03	6.89324	1203.84	219.924	8.97740E-03
m-Xylene	7.00172	1457.11	214.488	0.113916	7.00343	1458.21	214.609	0.119405
Cyclopentanone	6.07390	1453.46	216.082	6.49159E-04	6.06907	1450.04	215.666	8.26312E-04
Butyl acetate	6.01453	1360.76	203.123	8.36181E-04	6.02654	1368.05	203.930	1.13554E-03

[a] From refs. 6, 7, and 10.

present method is smaller than that obtained using the reported values of *A*, *B*, and *C*.

It is clear that the program described here provides a simple and accurate method for fitting data to the Antoine equation which is suitable for a microcomputer. Further, the method described here requires no initial estimation of the *A* or *B* values as input.

APPENDIX

```
10 DEFDBL A, B, C, L, P, R, S, T, W, X, Y, Z
20 '
30 'READ NUMBER OF DATA PAIR TO INPUT
40 READ NP
50 '
60 'DIMENSION ARRAYS
70 DIM LP(NP), P(NP), P1(NP), P2(NP), P3(NP), P4(NP),
       T(NP), W(NP)
80 '
90 'CLEAR VIDEO SCREEN
100 CLS
110 '
120 'READ TEMPERATURE, PRESSURE, WEIGHTING FACTOR
130 XX=0
140 FOR I=1 TO NP
150 READ T(I), P(I), W(I)
160 '
170 'APPLY DEMING'S PROCEDURE
180 W(I)=W(I)*(P(I)*2.3025850930)*(P(I)*2.3025850930)
190 '
200 XX=XX+W(I)
210 NEXT I
220 '
230 '
240 'COMPUTE DOUBLE PRECISION LOG (PRESSURE) VALUES
250 FOR I=1 TO NP
260 Z=P(I)
270 GOSUB 1330
280 LP(I)=(Z2/2.3025850930)
290 NEXT I
300 '
310 'READ INITIAL C VALUE, INCREMENT VALUE FOR C
320 READ R5, R9
330 R2=R5
340 '
350 'READ REPORTED A, B, AND C VALUES
360 READ QA, QB, QC
370 '
380 'READ NAME OF COMPOUND AND OUTPUT TO PRINTER
390 READ G$:LPRINTG$:LPRINT:LPRINT
400 '
410 '
420 'BEGIN MAIN PROGRAM, ITERATION ZERO
430 'INCREMENT C UNTIL EQUATION 16 YIELDS
440 'A POSITIVE NUMBER
450 '
460 A1=0:IT=0
470 '
480 'COMPUTE SUMMATIONS
490 GOSUB 930
500 '
```

```
510 'DISPLAY INTERMEDIATE RESULTS
520 GOSUB 1230
530 '
540 'WHILE EQ. 16 YIELDS NEGATIVE NUMBERS, LOOP TO 460
550 'SET "OLD A" EQUAL TO "NEW A"
560 IF R1#<0 THEN R5=R2: R2=R2+R9: R6=R1: A1=AN: IT=IT+1: GOTO 490
570 '
580 'ESTABLISH UPPER LIMIT FOR INTERVAL
590 R7=R2:R8=R1
600 '
610 'IF INITIAL C VALUE WAS TOO LARGE, DIVIDE THE
620 'CURRENT C VALUE BY 2 AND TRY AGAIN
630 IF A1=0 THEN R5=R5/2:R2=R5:R9=R5:GOTO 490
640 '
650 'TEST FOR CONVERGENCE
660 IF ABS(AN-A1)<1E-05 THEN 1430
670 '
680 'ITERATE ONCE AGAIN
690 IT=IT+1
700 '
710 'SET "OLD A" EQUAL TO "NEW A"
720 A1=AN
730 '
740 'DISPLAY INTERMEDIATE RESULTS
750 GOSUB 1230
760 '
770 'COMPUTE EQUATION 17
780 R2=R7-((R8/(R8-R6))*(R7-R5))
790 '
800 'COMPUTE SUMMATIONS
810 GOSUB 930
820 '
830 'IF EQ. 16 YIELDS A POSITIVE NUMBER, THEN
840 'RE-DEFINE THE UPPER LIMIT OF THE INTERVAL
850 IF R1>0 THEN R7=R2:R8=R1:GOTO 660
860 '
870 'EQ. 16 YIELDED A NEGATIVE NUMBER, RE-DEFINE
880 'LOWER LIMIT OF THE INTERVAL
890 R5=R2:R6=R1:GOTO 660
900 '
910 '
920 'COMPUTE SUMMATIONS
930 S1=0:S2=0:S3=0:S4=0:S5=0:S6=0
940 FOR I=1 TO NP
950 B=1/(R2+T(I))
960 S1=S1+B*W(I)
970 S2=S2+B*B*W(I)
980 S3=S3+B*B*B*W(I)
990 S4=S4+LP(I)*B*W(I)
1000 S5=S5+LP(I)*B*B*W(I)
1010 S6=S6+LP(I)*W(I)
1020 NEXT I
1030 '
1040 'COMPUTE EQUATION 13
1050 A=((S2*S5-S4*S3)/(S2*S2-S1*S3))
1060 '
1070 'COMPUTE EQUATION 14
1080 B=((XX*S5-S6*S2)/(S1*S2-XX*S3))
1090 '
1100 'COMPUTE EQUATION 16
1110 R1=(A*XX)-(S6+B*S1)
1120 '
1130 'SET "NEW A" EQUAL TO EQUATION 13
```

```
1140 AN=A
1150 '
1160 RETURN
1170 '
1180 '
1190 '
1200 'DISPLAY INTERMEDIATE RESULTS AFTER
1210 'EACH ITERATION
1220 '
1230 CLS:A!=A:B!=B:C!=R2
1240 PRINT"APPROXIMATE VALUES: ":PRINT"    "
1250 PRINT"A = ";A!
1260 PRINT"B = ";B!
1270 PRINT"C = ";C!
1280 PRINT"       ":PRINT"ITERATION #";IT
1290 RETURN
1300 '
1310 'LINES 1330 THROUGH 1400 COMPUTE DOUBLE
1320 'PRECISION LOG (PRESSURE) VALUES
1330 Z2=LOG(Z):I0=0:I2=SGN(Z2):IF I2<0 THEN Z=1/Z
1340 IF Z<1.065 THEN 1350 ELSE GOSUB 1390:Z=Z3:I0=I0+1:GOTO1340
1350 Z=(Z-1)/(Z+1):Z2=Z*Z:Z3=Z:I3=9:GOSUB1400:Z=Z+Z
1360 IF I0=0 THEN 1380
1370 FOR J=1 TO I0:Z=Z+Z:NEXT J
1380 Z2=Z*I2:RETURN
1390 Z3=SQR(Z):Z3=(Z3+Z/Z3)/2:Z3=(Z3+Z/Z3)/2:RETURN
1400 Z2=Z*Z:FOR K=3 TO ABS(I3) STEP2:Z3=SGN(I3)*Z3*Z2:Z=Z+Z3/K:
     NEXT:RETURN
1410 '
1420 ' COMPUTE BOILING POINT PER LEAST SQUARES CONSTANTS
1430 B1P=-((B/(2.8808136-A))+R2)
1440 '
1450 '
1460 ' COMPUTE BOILING POINT PER REPORTED CONSTANTS
1470 B2P=-((QB/(2.8808136-QA))+QC)
1480 '
1490 '

1500 GOSUB 1870
1510 LPRINT"       "
1520 LPRINT"       "
1530 '
1540 'PRINT BOILING POINTS
1550 LPRINT"EST. B.P. LSM > ";
1560 LPRINT USING"####.####";B1P
1570 LPRINT"EST. B.P. REP > ";
1580 LPRINT USING"####.####";B2P
1590 '
1600 A!=A:B!=B:C!=R2
1610 LPRINT"       "
1620 '
1630 'PRINT LEAST SQUARES CONSTANTS
1640 LPRINT"A COMPUTED > ";A!
1650 LPRINT"B COMPUTED > ";B!
1660 LPRINT"C COMPUTED > ";C!
1670 '
1680 LPRINT"       "
1690 '
1700 'PRINT REPORTED CONSTANTS
1710 LPRINT"A REPORTED > ";QA
1720 LPRINT"B REPORTED > ";QB
1730 LPRINT"C REPORTED > ";QC
1740 '
```

```
1750 LPRINT"       "
1760 '
1770 'PRINT SUM OF SQUARED ERRORS (SSE)
1780 LPRINT"SUM OF SQUARES LSM = ";
1790 LPRINT USING"#.####[[[[";CSSE
1800 LPRINT"SUM OF SQUARES REP = ";
1810 LPRINT USING"#.####[[[[";RSSE
1820 '
1830 'END PROGRAM
1840 END
1850 '
1860 'PRINT HEADINGS
1870 LPRINT"       "
1880 LPRINT"T(I)";
1890 LPRINT STRING$(8, 32);
1900 LPRINT"P(I)";
1910 LPRINT STRING$(8, 32);
1920 LPRINT"P-LSM";
1930 LPRINT STRING$(9, 32);
1940 LPRINT"DEL-P";
1950 LPRINT STRING$(5, 32);
1960 LPRINT"P-REP";
1970 LPRINT STRING$(9, 32);
1980 LPRINT"DEL-P REP"
1990 LPRINT"       "
2000 '
2010 'PRINT DATA, RESIDUALS, AND COMPUTE SSE
2020 FOR I=1 TO NP
2030 P1(I)=A-(B/(R2+T(I)))
2040 P1(I)=EXP(2.302585*P1(I))
2050 P2(I)=P1(I)-P(I)
2060 CSSE=CSSE+P2(I)*P2(I)
2070 '
2080 P3(I)=QA-(QB/(QC+T(I)))
2090 P3(I)=EXP(2.302585*P3(I))
2100 P4(I)=P3(I)-P(I)
2110 RSSE=RSSE+P4(I)*P4(I)
2120 '
2130 '
2140 'PRINT DATA AND RESIDUALS
2150 LPRINT USING"###.###";T(I);
2160 LPRINT STRING$(5, 32);
2170 LPRINT USING"###.###";P(I);
2180 LPRINT STRING$(5, 32);
2190 LPRINT USING"###.###";P1(I);
2200 LPRINT STRING$(5, 32);
2210 LPRINT USING"###.###";P2(I);
2220 LPRINT STRING$(5, 32);
2230 LPRINT USING"###.###";P3(I);
2240 LPRINT STRING$(5, 32);
2250 LPRINT USING"###.###";P4(I)
2260 NEXT I
2270 RETURN
2280 '
2290 'MERGE DATA FOLLOWING THIS LINE
2300 '
2310 '
2320 DATA 12
2330 DATA 66.698,8.586,1
2340 DATA 74.991,12.056,1
2350 DATA 80.908,15.127,1
2360 DATA 84.853,17.536,1
2370 DATA 96.299,26.362,1
```

```
2380 DATA 101.748,31.678,1
2390 DATA 108.096,38.933,1
2400 DATA 115.252,48.667,1
2410 DATA 122.359,60.166,1
2420 DATA 129.316,73.424,1
2430 DATA 136.104,88.494,1
2440 DATA 142.402,104.574,1
2450 DATA 200,100,6.069071,1450.044,215.6663
2460 DATA CYCLOPENTANONE
```

REFERENCES

1 G.W. Thomson, Chem. Rev., 38 (1946) 1.

2 W.N. Trump, Comput. Chem., 4 (1980) 117.

3 R.R. Dreisbach, Physical Properties of Chemical Compounds-I, Advances in Chemistry Series No. 15, American Chemical Society, Washington, DC, 1955.

4 R.R. Dreisbach, Physical Properties of Chemical Compounds-II, Advances in Chemistry Series No. 22, American Chemical Society, Washington, DC, 1959.

5 R.R. Dreisbach, Physical Properties of Chemical Compounds-III, Advances in Chemistry Series No. 29, American Chemical Society, Washington, DC, 1961.

6 E.F. Meyer and C.A. Hotz, J. Chem. Eng. Data, 21 (1976) 274.

7 E.F. Meyer and R.D. Hotz, J. Chem. Eng. Data, 18 (1973) 359.

8 D.R. Stull, Ind. Eng. Chem., 39 (1947) 517.

9 J. Timmermans, Physico-Chemical Constants of Pure Organic Compounds, Vol. 2, Elsevier, Amsterdam, 1965.

10 C.B. Willingham, W.J. Taylor, J.M. Pignocco and F.D. Rossini, J. Res. Natl. Bur. Stand., 35 (1945) 219.

11 W.E. Deming, Statistical Adjustment of Data, Wiley, New York, 1943.