SNARK-A FOUR-DIMENSIONAL TRENDSURFACE COMPUTER PROGRAM

ISOBEL CLARK
Department of Mineral Resources Engineering, Imperial College of Science and Technology,
Prince Consort Road,
London SW7 2BP, England

(Received 1 November 1975)

Abstract-A fast FORTRAN IV computer program for trend-surface analysis by the method of least squares is presented and discussed in detail. Applications in mining are mentioned briefly. A sample run using a set of data which is widely available is described.

Key Words: FORTRAN IV, Mining, Least squares, Trend-surface analysis.

INTRODUCTION

Trend-surface procedures of various types are used widely in the geosciences. This paper presents a computer program in FORTRAN IV for the evaluation of low-order polynomial trend surfaces by the method of least squares.

It is not considered within the scope of the paper to argue the case for or against the use of trend-surface analysis, and the reader is referred to Whitten (1956), Matheron (1971), and Davis (1973) for learned discussion of the merits and demerits of the analysis.

The program presented here was originally written in 1971 as a translation of the BALGOL program (Harbaugh, 1964) into FORTRAN IV. It has been modified since to increase speed and efficiency, and extended to include an analysis of variance and additional line-printer plotting options.

Although this program has been used by the author mainly for applications in mining, it was decided to provide an example of the analysis of a set of data which would be easily available to all potential users. The data set selected was the Fisher Iris Data (Fisher, 1936; Kendall and Stuart, 1966) which have been used widely in the past as a standard set of four variable measurements.

In the example, petal width (z) has been regressed on sepal length (w), sepal width (x) and petal length (y). Specimen input and output are given (Appendix 1), and an example of line-printer contour diagrams as a block constructed from the output.

APPLICATION IN MINING

Whitten (1966) states that trend surfaces are useful for two separate types of application in mining-those of exploration and exploitation. The author has found that in erratic ore bodies, such as the cassiterite veins encountered in Cornwall, low-order polynomial trend surfaces have
proved to be most useful in prediction of future payable ground, because although such lodes are sampled densely, the local variation, in values is so great as to mask the large-scale phenomenon of general trend in values. For this purpose the highest order of polynomial determined to be successful was a cubic. The computer program presented here calculates only linear, quadratic, and cubic.

Local grade values on the scale of stoping blocks may be evaluated by trend surfaces of a higher order. However, such surfaces tend to be unstable and reflect the behavior of erratic samples rather than the underlying ore body. Local estimation of ore reserves usually may be carried out more accurately by other statistical or geostatistical techniques.

Although the program contains approximately 800 statements, it occupies a relatively small amount of core and is comparatively fast. A large area under investigation may be split into neighborhoods and a trend surface produced for each of these. The neighborhoods then could be smoothed, as suggested by Pfältz (1973) and Whitten (1966), to produce a higher order surface without the instability of high-order polynomials. Alternatively, the neighborhood trend surfaces would be ideal as the first approximate surface required by the universal kriging technique of geostatistics.

**TREND-SURFACE COMPUTATION**

The notation in the following equations is consistent with that within the program. The three independent variables for sample point \( i \), which may or may not be geographical coordinates, will be denoted by \( w_i, x_i, \) and \( y_i \). The dependent variable is denoted by \( z_i \), and \( i \) takes values between 1 and \( n_s \). The lowest order of polynomial possible would be simply

\[
z_i = \bar{z} + \epsilon_i, \tag{1}\]

where \( \bar{z} \) is the arithmetic mean of the \( z_i \) values. This implies that there is no trend in z values which depends on the \( w, x, \) and \( y \) values, but merely random error about the overall mean. A linear trend surface, where the z value is directly (or linearly) proportional to each coordinate would be given by

\[
z_i = q_1 + q_2 w_i + q_3 x_i + q_4 y_i + \epsilon_i, \tag{2}\]

where \( q_1, q_2, q_3, \) and \( q_4 \) are known as the coefficients and \( \epsilon_i \) as the residual or the error term at point \( i \).

A second-order or quadratic surface would be described by

\[
z_i = s_1 + s_2 w_i + s_3 x_i + s_4 y_i + s_5 w_i^2 + s_6 w_i x_i + s_7 x_i y_i + s_8 y_i x_i + s_9 w_i^2 x_i + s_{10} y_i^2 + \epsilon_i, \tag{3}\]

where there are now 10 coefficients \( s_i \), and the error term \( \epsilon \).

The third-order surface used in this program is a partial cubic, where the cubic cross products have been eliminated to leave an equation with 13 coefficients and an error term \( \epsilon \).
The aim of the program is to determine those values of the coefficients of the trend-surface equation which reduce the so-called error term to a minimum. The criterion of a least-square solution is that the sum of squares of the residual terms should be a minimum.

That is

\[ \sum_{i=1}^{n} (e_i)^2 \]

should be minimized to determine the "best" solution for the values of

\[ f_1, f_2, \ldots, f_{13} \]

Using this criterion a set of simultaneous equations is evolved, which can be represented in matrix notation as

\[ \mathbf{f} = \mathbf{T}^{-1} \mathbf{r}, \]

where

\[ \mathbf{r} = [z_1, w_1, z_2, w_2, \ldots, z_n, w_n] \]

\[ \mathbf{f} = [f_1, f_2, \ldots, f_{13}] \]

and \( \mathbf{T} \) is given in Figure 1.
For the quadratic solution the simultaneous equations can be represented as
\[ s = T_{10} r_{10}, \]  
(6)
where \( T_{10} \) is taken as the first ten rows and columns of \( T \) (those enclosed by the dashed line in Fig. 1), and \( r_{10} \) contains the first ten elements of \( r \).

Similarly, the linear trend surface may be solved by minimizing \( \sum (\varepsilon_i)^2 \) in (2) giving
\[ q = T_4^{-1} r_4 \]  
(7)
where \( r_4 \) contains the first four elements of \( r \), and \( T_4 \) is as outlined by the dotted line in Figure 1.

Subroutine TOVE evaluates all those elements in \( T \) which are unique. For instance, because \( t_{1,6} = t_{2,4} = t_{4,2} = t_{6,1} \), only \( t_{6,1} \) is actually calculated and this value substituted in the other locations. This speeds calculation, although there is obviously a loss of generality. TOVE also calculates the elements of \( r \). Subroutine VORPAL is an adaptation of an IBM SSP routine for solution of sets of simultaneous equations of various sizes. The arguments of the subroutine—\( A, B, \) and \( M \)—correspond to the matrix of coefficients of the equations, the right-hand vector and the number of equations to be solved. For example, for the linear trend surface, the matrix of coefficients is \( T_4 \), the right-hand vector is \( Q \) and the number of equations is 4. To solve for the linear surface, the call becomes CALL VORPAL \((T_4, Q, 4, KS3)\). KS3 is returned with the value I if \( T_4 \) is singular. The coefficients of the linear trend surface are returned in \( Q \), and \( T_4 \) is destroyed in the process.

The value which the trend takes at any point (say \( g_0 \)) can be calculated by substitution of the three independent variable values at that point. The difference between the trend value and the observed value at any point is the residual value. If the user is prepared to make certain assumptions about these residuals, it is possible to test how "good" a fit the trend surface is to the data.

Firstly, the value of the residual at any point is assumed to be uncorrelated with the value of the trend at that point. Unless this is true, least squares should not be used strictly to estimate the coefficients of the trend surface. Now consider the variation in the sample values. This will be due to two (uncorrelated) causes

(a) the trend will cause variation between sample values in different parts of the deposit;
(b) "random" variation will give rise to local differences.

Using the first assumption it can be said that the variance of the \( z_i \) values is equal to the variance of the trend values at points \( i \), plus the variance of the residual terms at these points. That is
\[ \text{Var}(z_i) = \text{Var}(g_i) + \text{Var}(\varepsilon_i) \]
\[ \sum_{i=1}^{i=nz} (z_i - \bar{z})^2 = \sum_{i=1}^{i=nz} (g_i - \bar{g})^2 + \sum_{i=1}^{i=nz} (\varepsilon_i - \bar{\varepsilon})^2. \]  
(8)
Secondly, the average residual \( \bar{\varepsilon} \) is assumed to be zero, so that \( \bar{z} = \bar{g} \). This would imply that the trend surface gives an unbiased estimate of the \( z \) value at any point.

Then
\[ \sum_{i=1}^{i=nz} (z_i - \bar{z})^2 = \sum_{i=1}^{i=nz} (g_i - \bar{g})^2 + \sum_{i=1}^{i=nz} \varepsilon_i^2. \]  
(9)
Each of these terms can be calculated from the sample data and the trend equation. The usual practice is to calculate the left-hand term, usually called the "total sum of squares", and the last term or "residual sum of squares". The second term, or "trend sum of squares" then is determined by subtraction. However, when the number of samples in the analysis is large, this calculation can be time-consuming. The left-hand term can be calculated at the same time as \( T \) and \( r \) without extra effort, but the last term must be found by evaluating \( g_i \) at each point, subtracting \( z_i \) and summing the squares of the results. Recourse to the matrix notation shows that the trend sum of squares for the cubic surface can be evaluated easily as

\[
CSS = r'T = \sum_{i=1}^{13} r_{f_i},
\]

so that instead of calculating a cubic equation and squaring the result \( ns \) times, the second term is calculated with 13 multiplications.

A measure of the "goodness of fit" of a trend surface used widely is the percentage of the total sum of squares contributed by the trend. That is, the ratio of trend sum of squares to total sum of squares. This calculation has been included in the program but it is not considered reliable except for intuitive evaluation.

Suppose now, that the residual at any point can be regarded as a single sample from an approximately normal distribution. The trend surface will be said to be a "good" fit if the variance in sample values due to the trend is significantly greater than the variance due to random error. However, we shall not take the simple ratio between the two sums of squares. The total sum of squares is comprised of \( n (= ns) \) pieces of information \( z_1, z_2, z_3, \ldots, z_n \) minus the overall mean of the \( z_i \). Therefore it is said that the total sum of squares has \( n - 1 \) degrees of freedom. The trend sum of squares was calculated using thirteen coefficients and subtracting the overall mean \( \bar{z} \). Thus the trend sum of squares has twelve degrees of freedom. The residual sum of squares has \( n - 13 \) degrees of freedom because it is calculated from the total sum of the squares minus the trend sum of squares. These sums of squares may be referred to in the literature as "corrected sums of squares", because the overall mean \( \bar{z} \) is subtracted from the "uncorrected" sums of squares. However, in this paper the phrase "sums of squares" should be read as "corrected sums of squares".

If CSS represents (as in the program) the sum of squares explained by the cubic trend surface, and EM3 in the sum of squares of the residuals from the trend surface, then we calculate the following statistic

\[
F = \frac{CSS/12}{EM3/(n-13)}
\]

If the trend variance is no greater than the residual variance, then (11) should have an \( F \) distribution with 12 and \( n - 13 \) degrees of freedom. Tables of \( F \) values significant at various levels are available (Lindley and Miller, 1964). If the calculated \( F \) value is higher than the tabulated value for a selected level of significance, we can say that the cubic trend surface is a significantly better fit than equation (1).

Similarly, if QSS is the quadratic trend sum of squares, and EM2 the residual sum of squares from the quadratic, then

\[
F_{10} = \frac{QSS/9}{EM2/(n-10)}
\]
can be tested against tabulated values of $F$ with $9$ and $n - 10$ degrees of freedom; and for the linear, where SSS is the linear trend sum of squares and EMI the residual sum of squares

$$F_4 = \frac{\text{SSS}/3}{\text{EMI}/(n-4)}$$

should be $F$ distributed with $3$ and $n - 4$ degrees of freedom.

The trend surfaces also can be compared with each other. For instance, CSS-QSS will be the improvement in the sum of squares achieved by adding 3 terms to the trend equation. This value then will have 3 degrees of freedom, and

$$F_{32} = \frac{(\text{CSS-QSS})/3}{\text{EM3}/(n-13)}$$

should be an $F$ statistic with $3$ and $n - 13$ degrees of freedom. If (14) is higher than the tabulated value, we can say that the cubic equation is a significantly better fit than the quadratic. Similarly,

$$F_{61} = \frac{(\text{QSS-SSS})/6}{\text{EM2}/(n-10)}$$

will have $6$ and $n - 10$ degrees of freedom, and will indicate whether the quadratic trend surface is better than the linear surface.

Subroutine BOOJUM can produce a table of observed values, trend values and residuals for each data point. An analysis of variance table can be produced containing all the $F$-ratios described and the relevant information. If the ANOVA table is requested, the percentage of the corrected sum of squares explained by each surface will be printed. If any solution cannot be determined by VORPAL that part of the ANOVA will be omitted, and the trend value for each point given as zero.

SECTION PLOTTING OF TRENDS

The program has been designed to produce line-printer contour diagrams of profiles from a rectangular block in the three-dimensional space defined by $w$, $x$, and $y$. A range of values is defined for each of the three coordinates with respect to a block as shown in Figure 2.

![Figure 2. Orientation of plotted sections and blocks.](image)

The $w$ value at the top of the block need not be greater than that at the bottom. For instance, $w$ might be depth in an ore-reserve calculation. The user's $x$ coordinates may be measured from right to left, but this will not upset the program.
Subroutine TULGEY is set up to control and initiate plotting. The contours representing Z values are bands (not lines) by necessity. A single symbol will represent a range of Z values. This range is defined by CON — referred to as the contour interval, and a reference contour RF also is specified. This will be represented by an asterisk. Contours representing values below RF are shown as symbols A – L, and those above by M – X. Every alternate contour is represented by a blank. Although this may lead to ambiguity in some situations, it was shown that visual appreciation of the surface was enhanced.

Subroutine MOME prints out a table of Z values and the corresponding contour symbol. Subroutine TULGEY also calculates the predicted average Z value over the block defined for each surface. This can be useful particularly in estimation of average grade over large volumes. The arithmetic mean of the samples is given for comparison.

Profiles can be drawn with one particular coordinate held constant. If W is held constant, horizontal cross sections of the block are plotted, as if viewed from above. For Y held constant vertical cross sections are as viewed from in-front of the block.

The facility of printing contour plots of all six faces of the block at once also has been added (see Input information). This is simply done by a series of calls to UFFISH, the profile plotting routine, with appropriate arguments.

Table 1. Example of input to SNARK. Sample data were read from disc file

<table>
<thead>
<tr>
<th>Test run of standard SNARK on Fisher's iris data</th>
</tr>
</thead>
<tbody>
<tr>
<td>150034</td>
</tr>
<tr>
<td>(2X, A8, 4F5.1)</td>
</tr>
<tr>
<td>{if data were supplied on cards they would be inserted here.}</td>
</tr>
<tr>
<td>1101001100110011.0000.0000.0000.0000.0000.0000</td>
</tr>
<tr>
<td>7.5000 4.5000 4.0000 2.0000 6.0000 1.0000 1.0000 0.2500</td>
</tr>
<tr>
<td>6.0    3.0    3.5</td>
</tr>
</tbody>
</table>

One block and up to 8 profiles for each coordinate can be plotted for each surface. Options are provided for a block for each trend surface, and 8 profiles for each coordinate held constant and for each trend surface.

Plates 1 and 2 show a 10-in cube which the author determined useful for illustrating the variations of Z in three dimensions. The contours shown are those produced from the Fisher Iris Data (Fisher, 1963) with the Input as shown in Table 1.

Subroutine MIMSY prints out all relevant information about any block plotted, including the order of printing of the sides. Subroutine JUBJUB takes a vector of predicted Z values and converts these into a vector of symbols to be printed out as one line on the plot. A BLOCK DATA segment sets up the contour symbols in a nonstandard DATA statement.
The program will take up to 132 characters across the line. Only the width specified by the user is actually printed, so that the program may be used on a narrower printer by specifying a narrower width for the sections. Alternatively, the following changes can be made to accommodate larger or smaller printers. If the maximum printer width is NP characters (i.e. NP/10 in) then it is necessary to alter lines 460* and 461* (see Appendix 11) in subroutine TULGEY which check that the maps will fit on to the page. The four values of 12.9 on these lines should be altered to (NP-3)/10. For example, if the printer has 120 characters, 12.9 should be replaced by 11.7.

In subroutine JUBJUB, the dimensions of the arrays IV and ZI, on line 713*, should be changed to NP – 2. For example, with 120 print positions

713* DIMENSION IV(118), ZI(118).

In subroutine UFFISH, at line 747*, change the dimension of array Z to NP-2.

The program also assumes that the printer prints ten characters and six lines to the inch. This may be changed by altering lines 750* and 751* in subroutine UFFISH.

PROGRAM SPECIFICATION

Core and time requirements

All the information following relates to the use of the program on the CDC 6400 installation at Imperial College, London. The compiler used was the Minnesota FORTRAN Compiler MNF PSR 12. Compilation of the program and creation of a binary version occupied 21,250 words of core and took just under 5 sec.

The compiled binary file is loaded and run in 14,600 words of core. Several test runs were carried out to establish timings for various parts of the program. The solution of the three trend-surface equations, and the evaluation and printing of the comparison table of predicted values and residuals, both depend on the number of sample points used in the analysis. Table 2 shows some typical times for various sizes of sample sets. The analysis of variance table and the plotting times do not depend on the number of sample points in use. The ANOVA table is produced in 80 milliseconds, and the table giving details of contour symbols takes 40 milliseconds. Plotted contour surfaces depend on the order of the surface being plotted, and on the size of the plot required. The following times were obtained from test runs producing ten inch square contour plots

- Linear profile 925 msec
- Quadratic profile 1025 msec
- Cubic profile 1200 msec.

For comparison of core and time requirement, the Fisher Iris Data also was run with the standard package multiple regression program KWIKR8 (Esler, Smith, an Davis, 1968). This program will perform trend-surface analysis of up to quartic degree using three independent variables.

KWIKR8 compiles off the MNF PSR 12 compiler in 21,500 words, but for a set of 150 samples, requires 25,800 words of core store at run time. This represents an increase of about
75 percent on the requirements of the SNARK program. For 250 samples the storage requirements increase to almost 30,000 words, and for 1000 samples to about 59,000 words. Because SNARK does not store the data values internally, the core requirements remain at 14,600 words regardless of sample size. If SNARK were adapted for internal storage of data, the core requirements for a 1000 sample set would increase to approximately 21,000 words.

Table 2. Typical running time for different sized sample sets

<table>
<thead>
<tr>
<th>Number of samples</th>
<th>Time (msec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) Trend-surface</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>150</td>
</tr>
<tr>
<td></td>
<td>1500</td>
</tr>
<tr>
<td>(b) Table of</td>
<td>50</td>
</tr>
<tr>
<td>observed and</td>
<td>100</td>
</tr>
<tr>
<td>predicted</td>
<td>150</td>
</tr>
<tr>
<td>Z values</td>
<td>1500</td>
</tr>
</tbody>
</table>

To calculate the three surfaces: partial cubic, quadratic, and linear; to list the predicted values and residuals for each surface; and to calculate a few basic statistics for each surface (but no comparisons between surfaces), KWIKR8 requires 3.40 sec approximately, as opposed to a SNARK timing of 2.75 sec — an increase of over 20 percent in time requirements.

Thus, for a data set of the same size as the Fisher Iris Data, the standard package would entail an increase in costs of the order of 120 percent.

Other requirements

The data values are not stored within the program, but on a scratch disc file for the duration of the run. If no disc (or similar) facilities are available, the program can be adapted easily for internal data storage.

Input

Card 1: An 80 character title for the output from the run. Any characters available on the printer may be used.

Card 2: Columns 1-4 should give NS, the number of sample points to be used in the analysis. Column 5 contains zero if plotting is requested, 1 otherwise. Column 6 should contain 1 if calculation of the table of observed and expected values is desired; 2 if an analysis of variance is requested, 3 for both and zero for neither. Column 7 specifies the input channel number for the data. In this version 5 denotes card input, and 4 denotes disc or magnetic tape input.
Card 3: Specifies the format in which the data cards are to be read. The data are expected to be in the order:

W, X, Y, Z Identification

with each sample point occupying a separate card. The identification should be given in A format, up to A8.

Note: If this order for reading in the data is inconvenient to the user, only one line of the program needs to be changed.

If the data are on cards, these then follow.

Plot Card 1:
Column
1  zero if the six faces of the block are to be plotted for linear surface
2  zero for quadratic block
3  zero for cubic block
4  number of horizontal cross sections to be plotted (up to 8) – W held constant
5  zero if the linear plots at these values are not required
6  zero if the quadratic plots at these values are not required
7  zero if the cubic plots at these values are not required
8  number of vertical, left facing profiles to be plotted (up to 8) – X held constant
9  zero if linear plots not required
10 zero for no quadratic plots
11 zero for no cubic plots
12 number of vertical, front facing profiles to be plotted (up to 8) - Y held constant
13 zero for no linear plots
14 zero for no quadratic plots
15 zero for no cubic plots
16-20 height of finished block in inches (W axis)
21-25 width of finished block in inches (X axis)
26-30 depth of finished block in inches (Y axis).

Note that the width and depth of the block must be less than 13 in. The height of the block is unlimited.

Plot Card 2:
Each of the following occupies ten columns and must contain a decimal point.

Column
1-10  W value at the top of the block
11-20  W value at the bottom of the block
21-30  X value at the right hand side of the block
31-40  X value at the left hand side of the block
41-50  Y value at the back of the block
51-60  Y value at the front of the block
61-70  Reference contour value of Z, to be represented by *
71-80  Contour interval for Z values.

Plot Card 3a:
Only present if horizontal cross sections are requested. This card should contain those values of W for which profiles are requested. Each value is allowed ten columns and must contain a decimal point.

Plot Card 3b:
Only present if vertical, left-facing profiles are requested. This card should contain those values of X for which profiles are requested. Format as in 3a.

Plot Card 3c:
Only present if vertical, front-facing profiles are requested. This card should contain those values of Y for which profiles are requested. Format as in 3a.

Note: If no plotting is requested, no Plot Cards are necessary.

REFERENCES

Please note that, due to the inefficiency of the OCR program, Appendices have been omitted from this copy. If I cannot guarantee that it works, I won't hand it out. If you really want a copy of 25 year old Fortran code, please e-mail geoecosse@yahoo.co.uk