British Geological Survey

NATURAL ENVIRONMENT RESEARCH COUNCIL

Geoscience surface modelling: a summary of some mathematical methods

DGSM Programme

Internal Report IR/02/162

BRITISH GEOLOGICAL SURVEY

INTERNAL REPORT IR/02/162

Geoscience surface modelling: a summary of some mathematical methods

J C Loudon

Key words

Information and data resources; surface modelling; DGSM; interpolation; splining; kriging.

Bibliographical reference

LOUDON, J C. 2002. Geoscience surface modelling: a summary of some mathematical methods.. *British Geological Survey Internal Report,* IR/02/162. 21pp+ii.

Keyworth, Nottingham British Geological Survey 2002

BRITISH GEOLOGICAL SURVEY

The full range of Survey publications is available from the BGS Sales Desks at Nottingham and Edinburgh; see contact details below or shop online at www.thebgs.co.uk

The London Information Office maintains a reference collection of BGS publications including maps for consultation.

The Survey publishes an annual catalogue of its maps and other publications; this catalogue is available from any of the BGS Sales Desks.

The British Geological Survey carries out the geological survey of Great Britain and Northern Ireland (the latter as an agency service for the government of Northern Ireland), and of the surrounding continental shelf, as well as its basic research projects. It also undertakes programmes of British technical aid in geology in developing countries as arranged by the Department for International Development and other agencies.

The British Geological Survey is a component body of the Natural Environment Research Council.

Keyworth, Nottingham NG12 5GG

☎ 0115-936 3241 Fax 0115-936 3488
e-mail: sales@bgs.ac.uk
www.bgs.ac.uk
Shop online at: www.thebgs.co.uk

Murchison House, West Mains Road, Edinburgh EH9 3LA

☎ 0131-667 1000 Fax 0131-668 2683
e-mail: scotsales@bgs.ac.uk

London Information Office at the Natural History Museum (Earth Galleries), Exhibition Road, South Kensington, London SW7 2DE

☎ 020-7589 4090 Fax 020-7584 8270
☎ 020-7942 5344/45 email: bgslondon@bgs.ac.uk

Forde House, Park Five Business Centre, Harrier Way, Sowton, Exeter, Devon EX2 7HU

☎ 01392-445271 Fax 01392-445371

Geological Survey of Northern Ireland, 20 College Gardens, Belfast BT9 6BS

☎ 028-9066 6595 Fax 028-9066 2835

Maclean Building, Crowmarsh Gifford, Wallingford, Oxfordshire OX10 8BB

☎ 01491-838800 Fax 01491-692345

Parent Body

Natural Environment Research Council, Polaris House, North Star Avenue, Swindon, Wiltshire SN2 1EU

☎ 01793-411500 Fax 01793-411501
www.nerc.ac.uk

Foreword

This report was contributed by the author (james.loudon@physics.org) as the product of a work-experience attachment with the Digital Geoscience Spatial Modelling project at the British Geological Survey (BGS) during the summer of 2000. By assembling a brief account of some mathematical methods used to interpolate geological surfaces (created initially as a Web document) it may help to clarify the choice of alternatives and the underlying reasoning.

Acknowledgements

I wish to thank Dr P M Allen, who kindly agreed to the placement, and I F Smith, J L Laxton, K A Holmes and many other members of BGS staff for their generous help and guidance.

Contents

FIGURES

Summary

This is a brief account of some mathematics underlying various computer approaches to interpolating geoscience surfaces. A grid of data points is more readily modelled than scattered data and can be created from them on the basis of nearby points being more similar than distant ones. Kriging quantifies this approach. Methods for fitting gridded data include polynomial series, Fourier series or splines, using various criteria to define the best fit. Wavelets represent surfaces as a set of superimposed patterns on various scales, and show promise in data compression and pattern recognition.

1 Introduction

The purpose of surface modelling is to estimate from scattered data points the value of a variable at a point where the variable has not been measured. In the discussion that follows we take the variable to be the height of the surface but the conclusions can be applied to other variables such as the density of a substance in the rock. The surface model provides a smooth surface that goes through all the data points. It represents an averaged elevation and, as such, is smooth even though the actual surface is rough. The fitted surface should not introduce features for which there is no evidence and should not vary wildly between data points.

A good model should make some estimate of the likely uncertainty in the estimation of a surface. This uncertainty comes from several sources.

1. The errors inherent in the data.
2. The uncertainty in the estimation due to the distribution of data points.
3. The uncertainty caused by the roughness of the surface.
4. Discrepancies between the mathematical model and the real geology.

This document reviews some techniques currently in use as well as some that may be used in the future, giving a short description of each with a brief account of the background mathematics in smaller print.

2 Gridding

The ideal sampling pattern is a grid of points. It means that the data are evenly distributed over the area where predicted values are required and computer programmes only need to know the spacing between points rather than the position of every data point. Programmes written for gridded data run very much faster than those for a general distribution of points.

Therefore, nearly every computer program for fitting a surface through a set of points first draws a grid and estimates the height at each of the grid points using, not all of the data points, but the *nearby* points. Once the data has been gridded, a surface is then fitted through the gridded points. This two stage approach is rather unsatisfactory but hard to avoid. A map generated by this (and indeed any other) method should give some measure of the uncertainty in the fitted surface or indicate the positions of the data points so that a reader of the map may judge which parts of the map are most accurate.

Choosing the grid size is rather difficult. If it is very small, some cells will contain one point and most will contain none so any estimate at a grid point will be erroneous. It also cannot be very large for obvious reasons and a sensible compromise must be chosen.

There are many ways of estimating the values that should be assigned to the grid points. Simple methods include:

- Weighting the surrounding data points by the reciprocal of the distance to the grid points and taking the average.
- Taking the four cells that surround the grid point, finding the average height in each cell and taking the average of these four averages.

A more sophisticated method of assigning values to grid points is called kriging. This also gives an estimate of the uncertainty in estimating the grid points.

3 Kriging

Kriging treats the surface as a smooth trend on which a random variation is superimposed. The idea is this: We are trying to estimate the height at a point from the surrounding data points. We suppose that the point has a zone of influence and the data points within the zone have the same *mean* height and *standard deviation*. It is as though all the points within the zone of influence should have the same height on average and it is only by chance that they have assumed the configuration that we observe. Although it is odd to look at the heights of a hill in this way, it could be more reasonable when applied to, say, the percentage of gold in an ore body.

To estimate the height of this point, we say that it is the weighted sum of the data points around about it. We choose the weights to give a *Best Linear Unbiased Estimator*.

Best means that the expected error caused by fitting the surface in this manner is as small as possible.

Linear means that we have added the values of the data points not multiplied them.

Unbiased -- the weights are chosen so that we expect the fitted heights minus the true heights to be zero on average.

In order to do this, we need to know what the *correlogram* looks like. A correlogram is a plot of correlation against separation. The correlation is found from the following formula.

$$\text{covariance} = {}^{1}\!/_{N} \{ \sum_i (H(x_i) - m_x)(H(x_i+d) - m_{x+d})\}$$

and the correlation is the covariance divided by the standard deviation of x and the standard deviation of y.

What does all this mean? Well, N is the total number of data points, $H(x_i)$ is the height at position x_i and $H(x_i+d)$ is the height a distance d further along. m_x is the mean of the points in the zone of influence around x_i.

The correlogram looks something like this:

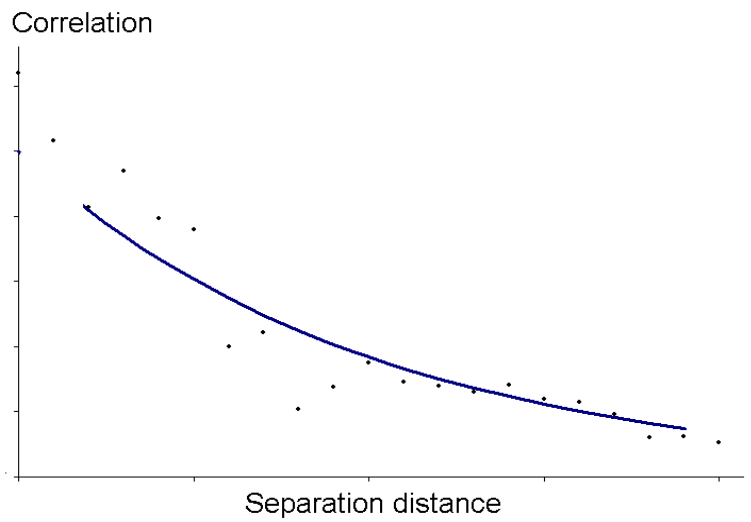

Figure 1: Example of correlogram

It gives a statistical measure of how similar the surface at one point is to the surface at another point. The smoother the surface, the less is the downward slope of the correlogram. In simple Kriging, the correlogram is assumed to be the same in every direction and for all locations but this may well be a false assumption as many surfaces are rough in one direction and smooth in another. There are more abstruse variations of Kriging that can deal with this.

Kriging

based on Isaacs and Srivastava, 1989

Kriging is a method for estimating the height at a position $\mathbf{x_0}$ from measured heights at *nearby* positions $\{\mathbf{x_i}\}$. *All* the heights are treated as *random variables* which follow the *same probability law*. The known heights (known outcomes of the random variable) are called $\{H(\mathbf{x_i})\}$ and the random variable $\mathcal{H}(\mathbf{x_0})$ describes the true, unknown height at $\mathbf{x_0}$. $\mathcal{H}(\mathbf{x_0})$ is estimated by the random variable $H(\mathbf{x_0})$ which is the weighted sum of the nearby known heights $\sum_i w_i H(\mathbf{x_i})$.

The weights are chosen so that the error $R(\mathbf{x}) \equiv H(\mathbf{x}) - \mathcal{H}(\mathbf{x})$ has an expected value of zero (condition 1) and a minimal variance (condition 2). That is, $H(\mathbf{x_0})$ is a best linear unbiased estimator of $\mathcal{H}(\mathbf{x_0})$.

The weights can then be expressed in terms of the covariance functions of H which must be estimated from the original data.

We now investigate the constraints imposed by conditions 1 and 2.

Condition 1

$$\mathrm{E}[R(\mathbf{x_0})] = 0$$

$$\Rightarrow \mathrm{E}[H(\mathbf{x_0}) - \mathcal{H}(\mathbf{x_0})] = 0$$

$$\Rightarrow \mathrm{E}\left[\sum_i w_i H(\mathbf{x_i}) - \mathcal{H}(\mathbf{x_0})\right] = 0$$

$$\Rightarrow \sum_i w_i \mathrm{E}[H(\mathbf{x_i})] = \mathrm{E}[\mathcal{H}(\mathbf{x_0})]$$

Now we assume that $\mathrm{E}[H(\mathbf{x_i})]$ are all the same and that $\mathrm{E}[H(\mathbf{x_i})] = E[\mathcal{H}(\mathbf{x_0})]$

So

$$\sum_i w_i = 1$$

Condition 2

$\mathrm{Var}[R(\mathbf{x_0})]$ must be minimised

We use the formula (derived in Appendix 2)

$$\mathrm{Var}\left[\sum_i w_i V_i\right] = \sum_i \sum_j w_i w_j \mathrm{Cov}[V_i V_j]$$

Now

$$\mathrm{Var}[R(\mathbf{x_0})] = \mathrm{Var}[H(\mathbf{x_0}) - \mathcal{H}(\mathbf{x_0})]$$

$$= \mathrm{Cov}[H(\mathbf{x_0})H(\mathbf{x_0})] - 2\mathrm{Cov}[H(\mathbf{x_0})\mathcal{H}(\mathbf{x_0})] + \mathrm{Cov}[\mathcal{H}(\mathbf{x_0})\mathcal{H}(\mathbf{x_0})]$$

The first term in the expression can be rewritten as

$$\mathrm{Cov}[H(\mathbf{x_0})H(\mathbf{x_0})] = \mathrm{Var}[H(\mathbf{x_0})] = \mathrm{Var}\left[\sum_i w_i H(\mathbf{x_i})\right]$$

$$\mathrm{Cov}[H(\mathbf{x_0})H(\mathbf{x_0})] = \sum_i \sum_j w_i w_j \mathrm{Cov}[H(\mathbf{x_i})H(\mathbf{x_j})]$$

The next term in the expression is

$$\begin{aligned}
2\mathrm{Cov}[H(\mathbf{x_0})\mathcal{H}(\mathbf{x_0})] &= 2\mathrm{Cov}\left[\sum_i w_i H(\mathbf{x_i})\mathcal{H}(\mathbf{x_0})\right] \\
&= 2\mathrm{E}\left[\sum_i w_i H(\mathbf{x_i})\mathcal{H}(\mathbf{x_0})\right] - 2\mathrm{E}\left[\sum_i w_i H(\mathbf{x_i})\right]\mathrm{E}\left[\mathcal{H}(\mathbf{x_0})\right] \\
&= 2\sum_i w_i \mathrm{Cov}[H(\mathbf{x_i})\mathcal{H}(\mathbf{x_0})]
\end{aligned}$$

And finally

$$\mathrm{Cov}[\mathcal{H}(\mathbf{x_0})\mathcal{H}(\mathbf{x_0})] = \mathrm{Var}[\mathcal{H}(\mathbf{x_0})] = \sigma^2$$

With the following compact notation

$$\sigma_R^2 = \mathrm{Var}[R(\mathbf{x_0})]$$

and

$$C_{ij} = \mathrm{Cov}[H(\mathbf{x_i})H(\mathbf{x_j})]$$

We have to minimise

$$\sigma_R^2 = \sigma^2 + \sum_i \sum_j w_i w_j C_{ij} - 2\sum_i w_i C_{i0}$$

With respect to the constraint $\sum_i w_i = 1$. We do this by using the method of Lagrange multipliers (see Appendix 1).

We first write $\sigma_R^2 = \sigma^2 + \sum_i \sum_j w_i w_j C_{ij} - 2\sum_i w_i C_{i0} + 2\mu(\sum_i w_i - 1)$ (note that this does not affect the equality as $\sum w_i - 1 = 0$).

We then differentiate this with respect to each of the w_i's and set each of the resulting equations equal to zero.

For example, if we differentiate with respect to w_l, we have

$$\frac{\partial \sigma_R^2}{\partial w_l} = 2\sum_i w_i C_{il} - 2C_{l0} + 2\mu = 0$$

or, rearranging

$$\sum_i w_i C_{il} + \mu = C_{l0}$$

All of these equations can be written as the matrix equation

$$\begin{pmatrix} C_{11} & \dots & C_{1N} & 1 \\ \vdots & & & \vdots \\ C_{N1} & \dots & C_{NN} & 1 \\ 1 & \dots & 1 & 0 \end{pmatrix} \begin{pmatrix} w_1 \\ \vdots \\ w_n \\ \mu \end{pmatrix} = \begin{pmatrix} C_{10} \\ \vdots \\ C_{N0} \\ 1 \end{pmatrix}$$

The weights can be found by inverting the symmetric matrix of covariances and there are many computer algorithms for this purpose.

Uncertainty

based on Isaacs and Srivastsva

The uncertainty in the fitted surface is the result of several effects:

1. The roughness of the surface

2. The clustering of the data points

3. The proximity of the data points to the point at which the surface is to be estimated.

These considerations are taken into account by the formula for the variance of an estimated point derived in the section on Kriging.

$$\sigma_R^2 = C_{00} + \sum_i \sum_j w_i w_j C_{ij} - 2 \sum_i w_i C_{i0}$$

The first term is the variance and this gives a measure of the roughness of the surface.

The second term is the covariance between individual points. The covariance gives a measure of the distance between points and so this term accounts for clustering.

The 3rd term is the covariance of the point we are estimating and the surrounding points. This gives a measure of the proximity to nearby points.

4 Orthogonal Series

Power series (polynomials) can be fitted to any distribution of data points. When additional terms (basis functions) are added, however, the entire equation must be recalculated. Orthogonal polynomials represent the series as a set of additive basis functions, which makes it easier to separate the fitted surface into component surfaces. The method described next is a way of fitting an orthogonal series to any arrangement of data points. It is, however, very computationally demanding, requiring n^3 operations where n is the number of data points, and so in practice is used only with small numbers of data points. However it has been suggested that this method could be used to grid data and then the surface could be fitted with more efficient techniques that are specific to gridded data. Orthogonal series, such as Fourier series or orthogonal polynomials, are normally calculated from gridded data.

Fitting an Orthogonal Series to Data

based on Riley, 1997 and Dixon, 1969

A surface can be represented by a sum of mathematical functions. The usual choice for these functions (called *basis functions*) are polynomials or sine waves (called a *Fourier series*). We shall call these functions $\phi_n(x, y)$ and the surface $f(x, y)$.

We want to express the surface as a sum of these basis functions, weighted by coefficients c_n.

$$f(x, y) = \sum_n c_n \phi_n(x, y)$$

In practice, $f(x, y)$ is unknown except at the data points where the surface has been measured. So, instead of a function, we write it as a column vector $\mathbf{F}$ which lists the data points (in any order). The functions $\{\phi_n(x, y)\}$ are known so we evaluate each of them at every data point and write each as a column vector $\mathbf{\Phi}_n$ (listing each point in the same order as $\mathbf{F}$).

So

$$\mathbf{F} = \sum_n c_n \mathbf{\Phi}_n \tag{1}$$

Now if we choose the $\mathbf{\Phi}_n$ vectors to be orthogonal that is

$$\mathbf{\Phi}_n . \mathbf{\Phi}_m = \left\{ \begin{array}{c} 0 \text{ for } m \neq n \\ |\mathbf{\Phi}_m|^2 \text{ if } m = n \end{array} \right\} \tag{2}$$

we can solve equation 2 if we take the dot product of both sides with $\mathbf{\Phi}_m$.

$$\begin{aligned} \mathbf{F} . \mathbf{\Phi}_m &= \sum_n c_n \mathbf{\Phi}_n . \mathbf{\Phi}_m \\ &= c_m |\mathbf{\Phi}_m|^2 \end{aligned}$$

So

$$c_m = \frac{\mathbf{F} . \mathbf{\Phi}_m}{|\mathbf{\Phi}_m|^2} \tag{3}$$

We have made the assumption that there are the same number of basis functions as data points. If we had more basis functions, not all of the c_n's could be found and there would be many surfaces that would fit through the same data points. This could be useful if there were evidence that the surface was not shaped in the manner the series fit suggested. Corrections could be made by (say) adding pseudo data points.

If we had fewer equations, there would be more than one answer for the same c_n and a fitted surface could not go through all the data points. If there were reason to think that the surface did not go through every data point a technique called *regression* could be used.

In fact, if we take the first few terms of a polynomial fit generated by this technique, this is a least squares regression fit of the surface.

In general, the basis vectors $\mathbf{\Phi}_n$ will not be orthogonal but we can create new basis vectors which are using the *Gram-Schmidt Orthogonalisation Procedure*.

The Gram-Schmidt Orthogonalisation Procedure

$$\begin{aligned} \mathbf{\Psi}_0 &= \mathbf{\Phi}_0 \\ \mathbf{\Psi}_1 &= \mathbf{\Phi}_1 - \frac{\mathbf{\Psi}_0 . \mathbf{\Phi}_1}{\mathbf{\Psi}_0 . \mathbf{\Psi}_0} \mathbf{\Psi}_0 \\ \mathbf{\Psi}_2 &= \mathbf{\Phi}_2 - \frac{\mathbf{\Psi}_1 . \mathbf{\Phi}_2}{\mathbf{\Psi}_1 . \mathbf{\Psi}_1} \mathbf{\Psi}_1 - \frac{\mathbf{\Psi}_0 . \mathbf{\Phi}_2}{\mathbf{\Psi}_0 . \mathbf{\Psi}_0} \mathbf{\Psi}_0 \\ &\text{etc...} \end{aligned}$$

5 Fourier Series

Fourier analysis breaks down a surface into sine (and cosine) waves. Figure 2 shows a square wave broken down into its components. The red line shows the square wave, the black line shows the first sine wave in the decomposition and the blue line shows the first four sine waves added together. The more sine waves that are added, the closer the Fourier representation becomes to the square wave. Figure 3 shows the sine waves that were added together to produce the blue line.

When the data is on a grid there is a very fast algorithm called a *Fast Fourier Transform* that breaks down a surface into components using $n \log_2 n$ operations. (See Numerical Recipies Online for more details.) Data compression is a means of storing the salient features of a surface and ignoring the irrelevancies. A Fourier representation can be used to do this -- one can compress the data by ignoring small sine waves. Unfortunately, sine waves stretch out to infinity but many surface features are localised. The wavelet transform splits the surface into localised functions called wavelets and can work as fast as a Fourier transform. However, the salient features of a surface are picked out by fewer wavelets compared with the number of sine waves required. This is why the wavelet transform is much better for data compression than Fourier analysis.

The Fourier transform also provides a plot of frequency against height (a *Fourier spectrum*). Taking sand dunes as an example, this tells you on average how high the dunes will be if they have a certain wavelength. It is deliberately designed to give no information about where the dunes are located. By contrast, the wavelet spectrum shows where each wavelength is located as well as its height.

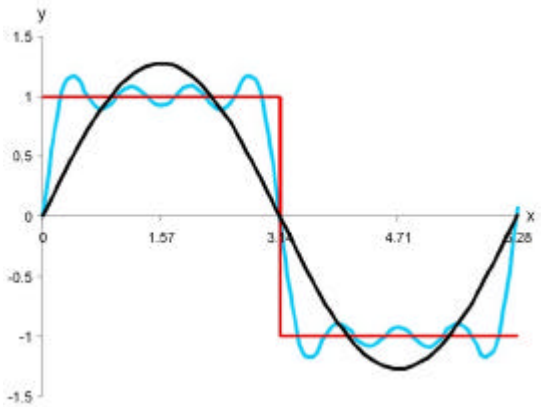

Figure 2: Fourier representation of a square wave

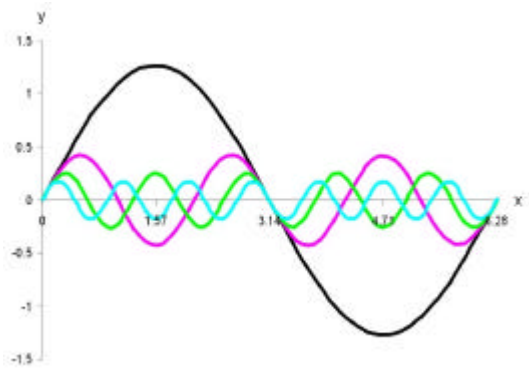

Figure 3: The sine waves added together in figure 2

Fourier Series

based on Riley, 1997 and Press, 1986

If $h(x)$ represents a line, a portion of the line from $x = -L/2$ to $x = L/2$ can be represented by the Fourier series

$$h(x) = \sum_{n=0}^{\infty} \left[c_n sin\left(\frac{2n\pi x}{L}\right) + d_n cos\left(\frac{2n\pi x}{L}\right)\right] \tag{1}$$

This can be written more conveniently with exponential notation

$$h(x) = \sum_{n=-\infty}^{\infty} H_n e^{i2\pi nx/L} \tag{2}$$

We can find H_k by multiplying both sides of 2 by $e^{-i2\pi mx/L}$ and integrating from $x = -L/2$ to $x = L/2$. Only the term in the sum where $m = n$ is left, the others integrate to zero.

$$\begin{aligned}
\int_{-L/2}^{L/2} h(x) e^{-i2\pi mx/L} dx &= \sum_{n=-\infty}^{\infty} \int_{-L/2}^{L/2} H_n e^{-i2\pi mx/L} e^{i2\pi nx/L} dx \\
&= \int_{-L/2}^{L/2} H_m e^{-i2\pi mx/L} e^{i2\pi mx/L} dx \\
&= H_m \int_{-L/2}^{L/2} dx \\
&= H_m L
\end{aligned}$$

So

$$H_m = \frac{1}{L} \int_{-L/2}^{L/2} h(x) e^{-i2\pi mx/L} dx \tag{3}$$

Discrete Fourier Transform

In reality $h(x)$ is only known at discrete points $\{x_k\}$ which we assume are separated by a length l. If we call the data points h_k and there are N of them then we can approximate (2) as

$$h_k = \sum_{n=0}^{N-1} H_n e^{i2\pi nx/L}$$

Since $L = Nl$ and $x_k = kl$ this can be written in a way that is independent of any lengths.

$$h_k = \sum_{n=0}^{N-1} H_n e^{i2\pi nk/N} \tag{4}$$

Similarly (3) can be written as

$$H_k = \frac{1}{N} \sum_{n=0}^{N-1} h_n e^{-i2\pi nk/N} \tag{5}$$

Equation (5) is called forward (Discrete) Fourier transform and equation (4) is the inverse Fourier transform. (Note the convention of starting the sums from $n = 0$ rather than ths symmetrical limits of (2) and (3). This can be done because the functions are periodic.)

6 Wavelets

Wavelets are localised functions that have an average of zero. There are many functions which satisfy this requirement and some are pictured below.

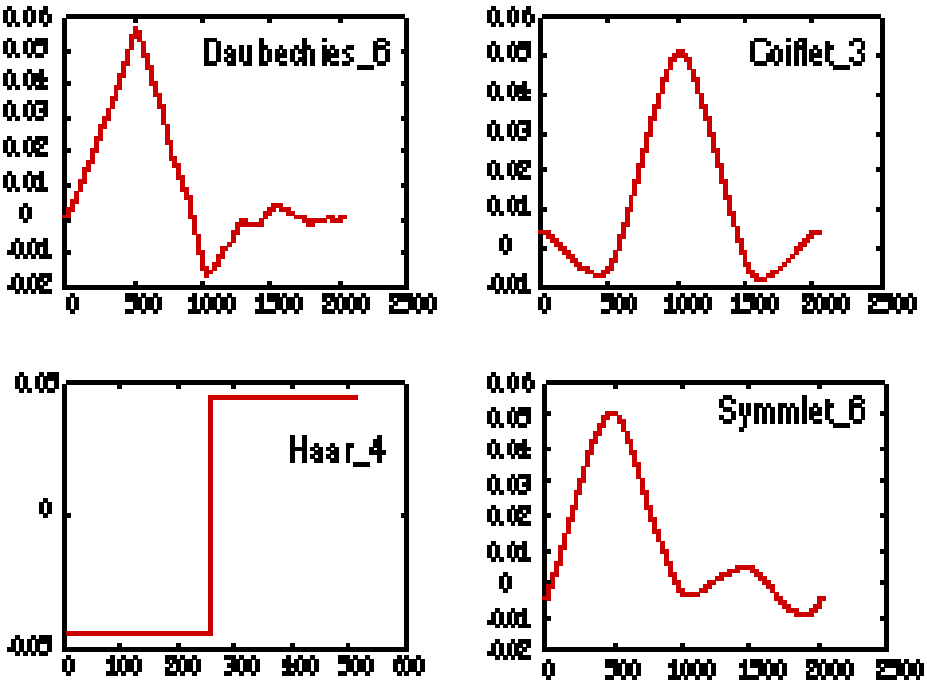

Figure 4: Examples of wavelets, reproduced with permission from Graps, 1997. © 1995 IEEE

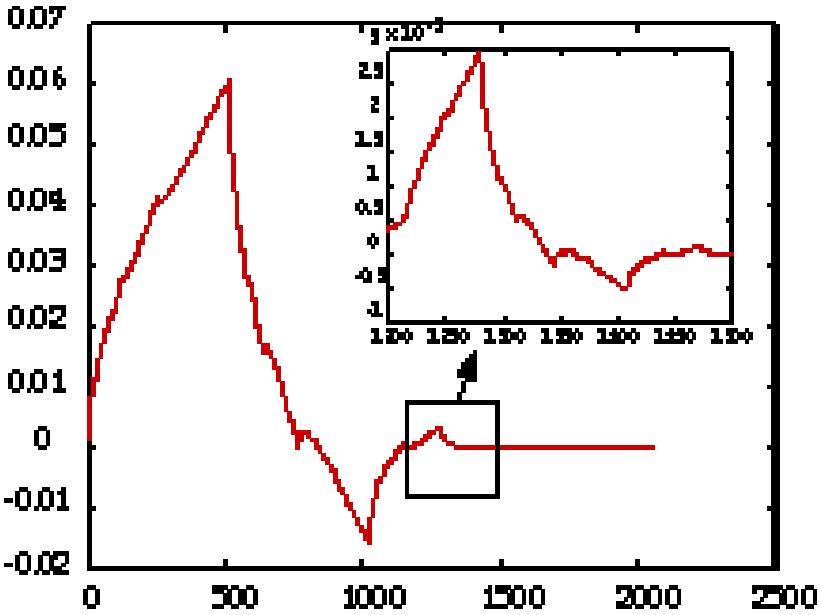

Figure 5: The Daubechies wavelet (a fractal) reproduced with permission from Graps, 1997. Note the similarity of the small enlarged portion to the original. © 1995 IEEE

Wavelet analysis breaks down a surface according to the scale of the features it contains. This decomposition is shown in the diagram below.

The red curve is the wavelet and the black is a cross section through a surface. The top row of the diagram shows a large wavelet analysing the surface which produces the blue curve on the top right. What happens is this: the wavelet is slid along the curve (the top row of the diagram shows it in 3 different positions). At every position, each point on the wavelet is multiplied by the point at the same position on the surface. The results of all these multiplications are added together and the answer is plotted on the top right hand diagram at the position of the centre of the wavelet. This gives one point on the blue diagram. The wavelet is then slid to the next position and the process is repeated. The result is the plot in blue.

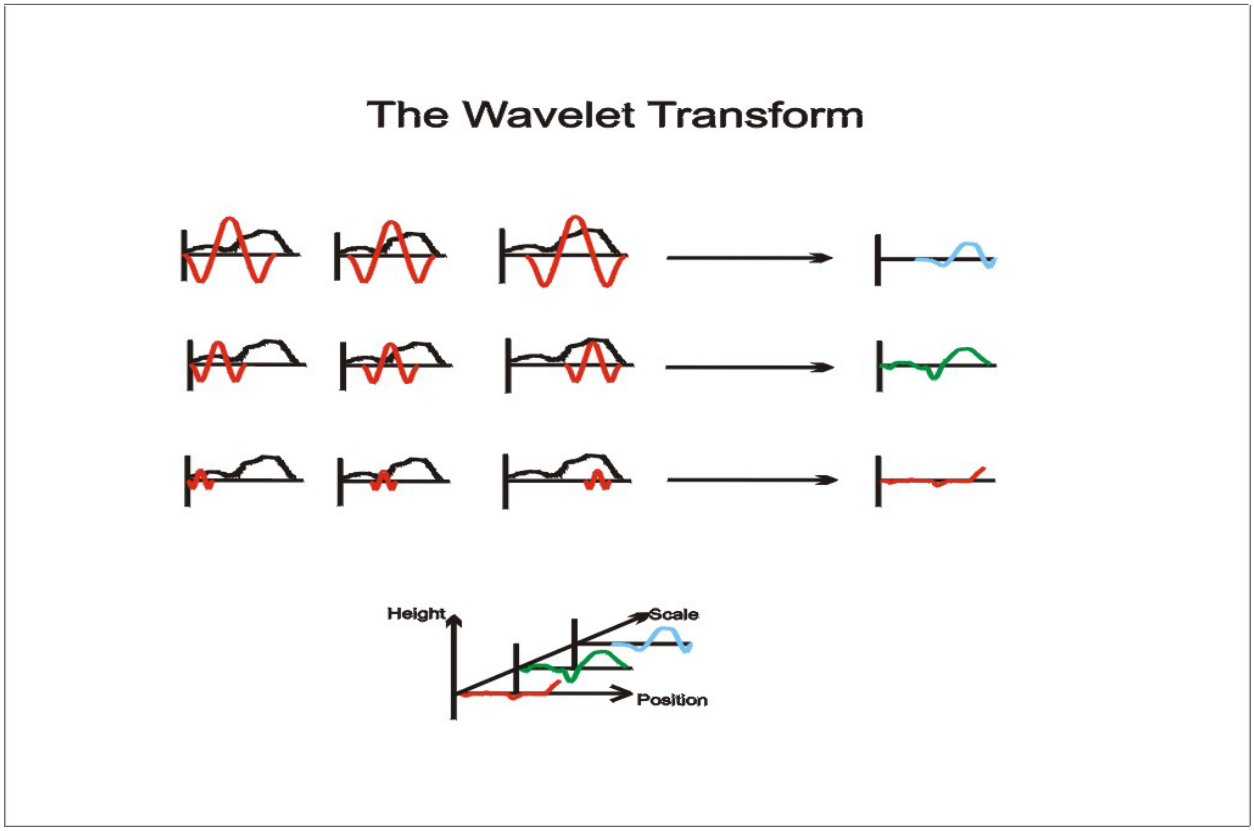

Figure 6: The wavelet transform, reproduced with permission from Graps, 1997. © 1995 IEEE

Having done this, the process is repeated with a smaller wavelet shown in the second row of the diagram. (The wavelet is scaled down in a way that preserves the aspect ratio.) This produces the green diagram. The final row shows analysis with a very small wavelet. The final result is usually plotted in a 3 dimensional diagram of position vs scale vs height as shown at the bottom.

Note the following:

1. The features that look most like the wavelet get the largest values.
2. Large wavelets pick out large scale features and small wavelets small scale features.
3. Very small wavelets pick out rapid changes in the surface.

Wavelets are currently used in data compression and signal processsing and many of its applications could be useful in geoscience. For example: It is useful to remove small scale roughness to show the overall trend of a surface. It can also be useful to remove the trend and observe the small scale features that would otherwise be obscured. Wavelets are well suited to this aim. Wavelets are widely used in data compression - a technique that is essential with

digitised maps. They are also used in signal processing for detecting sudden changes (edge detection). This could be useful for the detection of fault lines, etc.

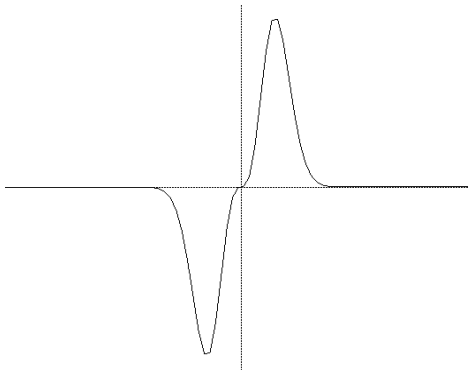

Figure 7: The third derivative of a Gaussian - a wavelet used for edge detection.

With the great range of shapes that wavelets exhibit, it may be possible to shape a wavelet to resemble a typical surface feature. Wavelet analysis could then pick out the possible locations of these features.

Wavelets

based on Polikar R.

The *correlation* of two functions compares one function with another. The correlation of $f(x)$ and $g(x)$ is defined to be

$$C(X) = f \otimes g = \int_{-\infty}^{\infty} f(x)g(x - X)\,\mathrm{d}x$$

The wavelet transform compares a function $f(x)$ with the *mother wavelet* $\phi(x)$ but it does so at different length scales. The mother wavelet is localised with a characteristic width L and so $\frac{1}{a}\phi\left(\frac{x}{a}\right)$ has a characteristic width aL.

Thus the wavelet transform of f is

$$\mathcal{W}[f] = \check{f}(X, a) = \int_{-\infty}^{\infty} \frac{1}{a}\phi\left(\frac{x - X}{a}\right) f(x)\,\mathrm{d}x$$

Properties of the Mother Wavelet

The mother wavelet must be localised and must have a mean of zero. To see why, we consider the mother wavelet in frequency space. If it is localised around a frequency ω_0 and has a width $\Delta\omega$, then dilating the wavelet by a in real space will cause $\omega_0 \to \frac{\omega_0}{a}$ and $\Delta\omega \to \frac{1}{a}\Delta\omega$. Thus the ratio $\frac{\Delta\omega}{\omega_0}$ = constant unless $\omega_0 = 0$. This leads to the requirement that the Fourier transform of $\phi(x)$ does not include a zero frequency component or, equivalently $\int_{-\infty}^{\infty} \phi(x)\,\mathrm{d}x = 0$, that is the mean is zero.

7 Splines

The popular *minimum tension* surface fit used by many computer packages for geoscience contouring is a spline fit. The spline fit is meant to mimic the behaviour of the bendable metal rule used by engineers to draw a smooth curve through a set of points. The engineer's spline assumes a shape of minimum tension (more precisely *tensile strain energy* since tension can be negative) which is the same as minimum curvature.

The method involves fitting cubic polynomials between pairs of points and at every point matching the slope and gradient. This produces a smooth curve that does not produce as many artefacts as the other types of fit. Sometimes it produces large humps in order to have a smooth surface. These can be reduced by relaxing the condition that the curvature is continuous to give a weighted spline. Artefacts are not necessarily a bad thing so long as they are recognised as such and do not obscure the natural features of the surface.

Splines

based on Lancaster and Salkauskas, 1986 and Roth, 1996

The spline curve mimics the behaviour of an engineer's spline – a metal rule which can be bent to go through a set of points $\{(x_i, Y_i)\}$ and assumes a shape which minimises its elastic strain energy which is proportional to the squared curvature $\left(\frac{\mathrm{d}^2 y}{\mathrm{d}x^2}\right)^2$.

There are, however, many ways in which a spline can go through a set of points depending on the length of the spline. To constrain the problem we fit 3rd degree polynomials $y_i = a_i + b_i u_i + c_i u_i^2 + d_i u_i^3$ between pairs of data points x_i and x_{i+1}. (To simplify the problem, we have used normalised coordinates $u_i \equiv \frac{x - x_i}{x_{i+1} - x_i}$.) The polynomials are chosen to satisfy the following constrains.

1. The spline goes through all the data points i.e. $y_i(0) = Y_i$

2. It is continuous i.e. $y_i(1) = y_{i+1}(0)$

3. The gradient is continuous i.e. $y_i'(1) = y_{i+1}'(0)$

4. The overall squared curvature (elastic strain energy) $\sum_i \int_{x_i}^{x_{i+1}} \left(\frac{\mathrm{d}^2 y}{\mathrm{d}x^2}\right)^2 \mathrm{d}x$ is minimal. (This is equivalent to minimising $C \equiv \sum_i \int_0^1 \left(\frac{\mathrm{d}^2 y_i}{\mathrm{d}u_i^2}\right)^2 \mathrm{d}u_i$).

Theorem – the curvature is continuous

We now show that condition 4 is equivalent to requiring that the curvature is continuous.

We use the method of Lagrange multipliers to minimise C with respect to conditions 1, 2 and 3.

$$C = \sum_i \left[\int_0^1 \left(\frac{\mathrm{d}^2 y_i}{\mathrm{d}u_i^2}\right)^2 du_i + \lambda_i (y_i'(1) - y_{i+1}'(0)) + \mu_i (y_i(1) - y_{i+1}(0)) \right]$$

i.e.

$$\begin{aligned} C = \sum_i [& 4c_i + 12 c_i d_i + 12 d_i \\ & + \lambda_i (2c_i + 3d_i + b_i - b_{i+1}) \\ & + \mu_i (a_i - a_{i+1} + b_i + c_i + d_i)] \end{aligned} \tag{1}$$

Minimising with respect to a_i, b_i, c_i and d_i,

$$\frac{\mathrm{d}C}{\mathrm{d}c_i} = 8c_i + 12 d_i + 2\lambda_i + \mu_i = 0 \tag{2}$$

$$\frac{\mathrm{d}C}{\mathrm{d}d_i} = 12 c_i + 24 d_i + 3\lambda_i + \mu_i = 0 \tag{3}$$

$$\frac{\mathrm{d}C}{\mathrm{d}a_i} = \mu_i - \mu_{i-1} = 0 \tag{4}$$

$$\frac{\mathrm{d}C}{\mathrm{d}b_i} = \lambda_i - \lambda_{i-1} + \mu_i = 0 \tag{5}$$

Using equation (5) to get rid of μ_i in equations (2) and (3) and then subtracting (2) from (3) to get rid of λ_{i-1} gives

$$\lambda_i = -4[c_i + 3d_i] \tag{6}$$

and substituting this back into (2) gives

$$\lambda_{i-1} = -4c_i$$

But (6) tells us that

$$\lambda_{i-1} = -4(c_{i-1} + 3d_{i-1})$$

Equating these two and dividing by -2 gives

$$2c_i = 2c_{i-1} + 6d_{i-1}$$

And therefore

$$y_i''(0) = y_{i-1}''(i-1)$$

So the curvature is continuous *Q.E.D.*

Finding the Coefficients of a Spline

We use the conditions that the line, the gradient and the curvature are continuous to find the coefficients a_i, b_i, c_i, d_i. If there are N line segments, we need to fit 4 coefficients to each and so have $4N$ unknowns. Labelling the data points (x_i, Y_i), we have the following equations for each interior data point.

$$\begin{aligned} y_{i-1}(1) &= Y_i \\ y_i(0) &= Y_i \\ y_{i-1}'(1) &= y_i'(0) \\ y_{i-1}''(1) &= y_i''(0) \end{aligned}$$

This gives us $4(N-1)$ equations. We also have

$$\begin{aligned} y_0(0) &= Y_0 \\ y_{N-1}(1) &= Y_n \end{aligned}$$

Which gives a total of $4N-2$ equations. To fully constrain the curve we require that the curvature vanishes on the boundary.

$$\begin{aligned} y_0''(0) &= 0 \\ y_{N-1}''(1) &= 0 \end{aligned}$$

To find the coefficients, we note that

$$\begin{aligned} y_i(0) &= a_i \\ y_i(1) &= a_i + b_i + c_i + d_i \\ y_i'(0) &= b_i \equiv D_i \\ y_i'(1) &= b_i + 2c_i + 3d_i \equiv D_{i+i} \end{aligned}$$

And, solving

$$\begin{aligned} a_i &= Y_i \\ b_i &= D_i \\ c_i &= 3(Y_{i+1} - Y_i) - 2D_i - D_{i+1} \\ d_i &= 2(Y_i - Y_{i+1}) + D_i + D_{i+1} \end{aligned}$$

It only remains to find D_i. From the continuity of interior curvature we have

$$2c_{i-1} + 6d_{i-1} = 2c_i$$

$$\Rightarrow D_{i-1} + 4D_i + D_{i+1} = 3(Y_{i+1} - Y_{i-1})$$

And, from setting the curvature to be zero at the end points

$$2D_0 + D_1 = 3(Y_1 - Y_0)$$

$$D_{N-1} + 2D_N = 3(Y_N - Y_{N-1})$$

Which can be written as the matrix equation

$$\begin{pmatrix} 2 & 1 & & & & \\ 1 & 4 & 1 & & & \\ & 1 & 4 & 1 & & \\ & & & \ddots & & \\ & & & 1 & 4 & 1 \\ & & & & 1 & 2 \end{pmatrix} \begin{pmatrix} D_0 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ D_N \end{pmatrix} = \begin{pmatrix} 3(Y_1 - Y_0) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ 3(Y_N - Y_{N-1}) \end{pmatrix}$$

This can be solved by inverting the symmetric, tridiagonal matrix – a well known problem for which there are many computer algorithms.

For a rectangularly gridded surface, this technique can be directly generalised to 3 dimensions by fitting splines along the lines of data points in both directions.

Appendix 1

based on Riley, 1997

The Method of Lagrange Multipliers

We introduce the method by considering the following example: suppose we are walking around a hill whose height varies according to the function $f(x, y)$. For simplicity we shall assume that it is a rather idealised hill which rises smoothly to its summit and has no other peaks or valleys – that is, the function f has only one stationary point - the top of the hill. The maximum height of the hill can be found by considering a small change in the height $\mathrm{d}f$. If the value of f is unchanged by this small step, we are at a stationary point – the top of the hill. We can then find the position of the summit by expanding f by the chain rule.

$$\mathrm{d}f = \frac{\partial f}{\partial x}\mathrm{d}x + \frac{\partial f}{\partial y}\mathrm{d}y = 0 \tag{1}$$

There is no function relating the steps $\mathrm{d}x$ and $\mathrm{d}y$ - they can be made completely independently of one another. Choosing $\mathrm{d}y$ to be zero and $\mathrm{d}x$ to be non zero implies that

$$\frac{\partial f}{\partial x} = 0 \tag{2}$$

Similarly, choosing $\mathrm{d}x = 0$ and $\mathrm{d}y = 0$, we have

$$\frac{\partial f}{\partial y} = 0 \tag{3}$$

The point is that equation 1 must be true for *all* small steps $\mathrm{d}x$ and $\mathrm{d}y$ and so equations 2 and 3 must be true *simultaneously*.

The method of Lagrange multipliers is used when we wish to maximise $f(x, y)$ subject to the constraint $g(x, y) =$ constant. In our example, this corresponds to having a path over the hill and we wish to know the maximum height the path reaches (the path does not necessarily cross the summit). The equation describing the location of the path $g(x, y) =$ constant relates x and y and so we cannot say that $\mathrm{d}x$ and $\mathrm{d}y$ can be chosen independently. Now, we could rearrange the equation of the path so that we have y equals some function of x and substitute this for y so that f would be a function of x only. We could then maximise f with respect to x only and this would solve the problem.

However, g is frequently a complicated function and rearranging it is difficult. The method of Lagrange multipliers makes this simpler. We want to find where

$$\mathrm{d}f = \frac{\partial f}{\partial x}\mathrm{d}x + \frac{\partial f}{\partial y}\mathrm{d}y = 0$$

and since $g(x, y) =$ constant any change in g, $\mathrm{d}g = 0$. So

$$\mathrm{d}g = \frac{\partial g}{\partial x}\mathrm{d}x + \frac{\partial g}{\partial y}\mathrm{d}y = 0 \tag{4}$$

We now multiply 4 by an as yet undetermined multiplier λ (a so-called *Lagrange* multiplier) and add it to 3.

Thus

$$\mathrm{d}(f + \lambda g) = \left(\frac{\partial f}{\partial x} + \lambda\frac{\partial g}{\partial x}\right)\mathrm{d}x + \left(\frac{\partial f}{\partial y} + \lambda\frac{\partial g}{\partial y}\right)\mathrm{d}y = 0$$

We can now choose λ to make $\mathrm{d}x$ and $\mathrm{d}y$ independent. This implies

$$\frac{\partial f}{\partial x} + \lambda\frac{\partial g}{\partial x} = 0 \tag{5}$$

and

$$\frac{\partial f}{\partial y} + \lambda \frac{\partial g}{\partial y} = 0 \tag{6}$$

So solving 5 and 6 together with the constraint equation $g(x, y) = \text{constant}$ we find the location of the stationary point. (We also find the value of λ required to render $\mathrm{d}x$ and $\mathrm{d}y$ independent but usually this is not very interesting.)

So instead of maximising f subject to g, we maximise $f + \lambda g$ unconstrainedly.

Appendix 2

based on Crawshaw, 1986

Proof of the Formula Quoted in Kriging

In the discussion on kriging, the following formula was quoted.

$$\mathrm{Var}\left[\sum_i w_i V_i\right] = \sum_i \sum_j w_i w_j \mathrm{Cov}[V_i V_j]$$

We now prove the above result.

$$\begin{aligned}
\mathrm{Var}\left[\sum_i w_i V_i\right] &= \mathrm{E}\left[\left(\sum_i w_i V_i\right)^2\right] - \left(\mathrm{E}\left[\sum_i w_i V_i\right]\right)^2 \\
&= \mathrm{E}\left[\sum_i w_i^2 V_i^2 + \sum_i \sum_{i \neq j} w_i V_i w_j V_j\right] - \mathrm{E}\left[\sum_i w_i V_i\right]^2 \\
&= \mathrm{E}\left[\sum_i \sum_j w_i w_j V_i V_j\right] - \mathrm{E}\left[\sum_i w_i V_i\right] \mathrm{E}\left[\sum_j w_j V_j\right] \\
&= \sum_i \sum_j w_i w_j (\mathrm{E}[V_i V_j] - \mathrm{E}[V_i]\mathrm{E}[V_j]) \\
&= \sum_i \sum_j w_i w_j \mathrm{Cov}[V_i V_j]
\end{aligned}$$

Q.E.D

8 References

Dixon R. 1969. *Orthogonal Polynomials as a Basis for Objective Analysis*. Meteorological Office, Scientific Paper No. 30 (London, Her Majesty's Stationery Office)

Graps, A. 1995. An introduction to wavelets. *IEEE Computational Science and Engineering.* Vol. **2**(2), pp50-61.

Graps, A. 1997. An introduction to wavelets. http://www.amara.com/IEEEwave/IEEEwavelet.html

Isaaks E H and Srivastava R M. 1989. *An Introduction to Applied Geostatistics.* (Oxford, Oxford University Press)

Lancaster P and Salkauskas K. 1986. *Curve and Surface Fitting.* (London, Academic Press)

Polikar, R. 2001. The Engineer's Ultimate Guide to Wavelet Analysis: the Wavelet Tutorial. http://engineering.rowan.edu/~polikar/WAVELETS/WTtutorial.html

Press W H, Teukolsky S A, Vetterling W T and Flannery, B P, 1986. *Numerical Recipies in FORTRAN*. (Cambridge, Cambridge University Press)

Press, W H et al. 2002. Numerical Recipes On-line. http://www.ulib.org/webRoot/Books/Numerical_Recipes/

Riley K F, Hobson M P and Bence S J. 1997. *Mathematical Methods for Physics and Engineering.* (Cambridge, Cambridge University Press)