Imperfectly Separable Models

Jon F. Claerbout

Textbooks, particularly those on partial differential equations and those on probability, are filled with *separable* models of reality. The real world contains many data bases which are roughly separable. But poor agreement between theory and practice can often be traced to sensitivity of results to departure from perfect separability. This paper describes the problem in general, shows an effective technique to deal with it, and then cites specific examples.

The Simplest Separable Model

The simplest separable model of a two dimensional data set d_{ij} is

$$d_{ij} = a_i + b_j \tag{1}$$

The data matrix is assumed to be separable into a sum of a column vector and a row vector. Instead of having the data decompose into a *sum* of two parts, another very interesting form of separability is to have it decompose into a *product* of two parts. Product separability is often related to summation separability by taking logarithms of all quantities. But if the data d_{ij} is negative or can be complex, the use of logarithms is not so simple and may not work at all. Complex values arise in practice where a sequence of linear filters create the data. We will return to product separability at the very end of this paper, thus limiting ourselves mainly to the additive model.

In equation (1) it is stated that a_i is not a function of j and that b_j is not a function of i. Reality might be that a_i should be a *slowly* variable function of j. For example, perhaps i spans the geographical range of a marine hydrophone cable and j spans the length of the survey, a much greater distance. Almost everything about a geophysical data set has a perceptible regional j variation.

The Running Mean

Recall that the simple act of computing an average amounts to finding the solution to a least squares problem. To see this, define a quadratic form

$$Q(a) \quad = \quad \sum_{i=1}^{N} (d_i - a)^2 \tag{2}$$

To minimize Q we set to zero the derivative with respect to a. This gives the average

$$a \quad = \quad \frac{1}{N} \sum_{i=1}^{N} d_i \tag{3}$$

Next consider a data set d_i with the property that its mean value a should be considered to be a slowly variable function of i. Everyone has an intuitive idea of how such a *running mean* could be computed. Let us find least squares problems whose solutions are running means. The basic way to get a smooth running mean a_i is to construct a penalty function Q which increases with the roughness of a_i. For instance, a roughened mean could be defined by

$$\mathbf{B\,a} \quad = \quad \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \tag{4}$$

A penalty function is a scalar, which we will form by taking the dot product of the vector (4) on itself. Such a dot product is a sum of squares. We could weight each term the same, or we could weight them all differently with a diagonal matrix of weights, say **W**. Thus in terms of matrix algebra notation, the quadratic scalar to be minimized will be

$$Q \quad = \quad \mathbf{(d-a)}^* \mathbf{(d-a)} \; + \; \mathbf{a}^* \mathbf{B}^* \mathbf{W B\,a}$$

For simplicity sake we will make the definition $\mathbf{T} = \mathbf{B}^* \mathbf{W}\, \mathbf{B}$. Thus

$$Q \quad = \quad \mathbf{(d-a)}^* \mathbf{(d-a)} \; + \; \mathbf{a}^* \mathbf{T\,a} \tag{5}$$

Books on optimization theory, such as FGDP, show that the minimization can be achieved by setting to zero the derivative with respect to $\mathbf{a}^*$. Thus

$$\mathbf{(I + T)\,a} \quad = \quad \mathbf{d} \tag{6}$$

Computationally we have to solve a tridiagonal set of simultaneous equations for the a_i. In practice, this is easy to do. The cost increases only linearly with the size of the matrix. The solution technique is described in detail elsewhere (FGDP for example).

Quadratic Form for the Imperfectly Separable Problem

Consider next the quadratic form

$$Q(\mathbf{a},\mathbf{b}) \;=\; (\mathbf{d}-\mathbf{a}-\mathbf{b})^* (\mathbf{d}-\mathbf{a}-\mathbf{b}) \;+\; \mathbf{a}^* \mathbf{T}_x \mathbf{a} \;+\; \mathbf{b}^* \mathbf{T}_y \mathbf{b} \tag{7}$$

Before specific definition of $\mathbf{T}_x$ and $\mathbf{T}_y$ we proceed to the minimization. We set to zero the partial derivatives.

$$0 \;=\; \frac{\partial Q}{\partial \mathbf{a}^*} \;=\; -(\mathbf{d}-\mathbf{b}-\mathbf{a}) \;+\; \mathbf{T}_x \mathbf{a} \tag{8a}$$

$$0 \;=\; \frac{\partial Q}{\partial \mathbf{b}^*} \;=\; -(\mathbf{d}-\mathbf{b}-\mathbf{a}) \;+\; \mathbf{T}_y \mathbf{b} \tag{8b}$$

Equations (8a) and (8b) are a set of simultaneous equations for $\mathbf{a}$ and $\mathbf{b}$ which could be be solved analytically. However, we will not do so because the solution is expressed in terms of inverses of matrices which will usually be too big to invert.

The desired generalization of (1) to allow a to be a smooth function of j and b to be a smooth function of i is included as a special case of (7). For example, (7) can be specialized to

$$Q(a_{i,j},b_{i,j}) \;=\; \sum_{i,j}(d_{i,j}-a_{i,j}-b_{i,j})^2 \;+\; \sum_{i,j}(a_{i,j}-a_{i,j-1})^2 \;+\; \sum_{i,j}(b_{i,j}-b_{i-1,j})^2 \tag{9}$$

To equate (9) to (7), the elements of the matrix d_{ij} must be packed into the vector $\mathbf{d}$. Taking d_{ij} to be an $N\times N$ matrix, it means that the vector $\mathbf{d}$ contains N^2 elements. The $\mathbf{T}$ matrices contain N^4 elements. Equation (4) specifies a case with $N=4$. Taking $N=4$ in both i and j directions, the operators $\mathbf{T}_x$ and $\mathbf{T}_y$ are 16×16 matrices. It is a simple exercise to work out the placement of the elements. But there is no real reason to do so. In most cases of interest, N is of the order of a hundred or a thousand. Hence N^2 is about 100,000 and the number of elements in the matrices is about 10^{10}. Thus the direct solution method is impractical. There are, however, many iterative methods. Ordinarily we have no intent of iterating to convergence and we often think in terms of fewer than three iterations. The simplest iterative method is considered next.

Define the residual $\mathbf{r}$ by

$$\mathbf{r} \;=\; \mathbf{d}-\mathbf{a}-\mathbf{b} \tag{10}$$

Rearrange (8)

$$(\mathbf{I} + \mathbf{T}_x)\,\mathbf{a}_{next} \quad = \quad \mathbf{r} + \mathbf{a}_{previous} \tag{11a}$$

$$(\mathbf{I} + \mathbf{T}_y)\,\mathbf{b}_{next} \quad = \quad \mathbf{r} + \mathbf{b}_{previous} \tag{11b}$$

The operator $(\mathbf{I}+\mathbf{T})^{-1}$ may be called a low pass filter. In words, (11a) simply says that the new **a** is a low pass filtered version of the residual plus the old **a**. This low pass filtering is done independently for each value of the other coordinate (j or y). A simple solution technique is to iterate back and forth between (11a) and (11b). (Naturally, the residual (10) is updated between steps.) This technique is known to converge if the **T** matrices are positive definite. When convergence is achieved then (8a) says that

a = Lowpass(data less all model parameters but **a**)

Two Dimensionally Smooth Fitting Functions.

The solutions obtained in the previous section are smooth in one direction, but not the other. It is not difficult to generalize the techniques so as to provide controllable filtering in both directions. For example (11a) could be replaced by something like

$$(\mathbf{I} + \mathbf{T}_y)(\mathbf{I} + \mathbf{T}_x)\mathbf{a}_{next} \quad = \quad \mathbf{r} + \mathbf{a}_{previous} \tag{12}$$

A little algebra shows that this would be providing a minimization of the quadratic form

$$Q(\mathbf{a}) \quad = \quad (\mathbf{d}-\mathbf{a})^*\,(\mathbf{d}-\mathbf{a}) \;+\; \mathbf{a}^*\,[(\mathbf{I}+\mathbf{T}_y)(\mathbf{I}+\mathbf{T}_x)-\mathbf{I}]\,\mathbf{a} \tag{13}$$

Near Surface Modeling

Analysis of surface statics commonly proceeds by picking some travel times of a marker bed and then fitting with an equation like

$$t_{sg} \quad \approx \quad a_s + b_g + c_{\frac{g+s}{2}} + d_{\frac{g+s}{2}}\,(g-s)^2 \tag{14}$$

Turhan Taner and also Larry Morley analyzed near surface spectral anomalies by computing the spectrum of each trace, then taking logarithms to convert the convolutive or multiplicative problem to an additive one. The log spectra are analyzed with the equation

$$d_{sg} \quad \approx \quad a_s + b_g + c_{\frac{g+s}{2}} + f_{\frac{g-s}{2}} \tag{15}$$

Let us consider the applicability of the methods of this paper to the models (14) and (15). Instead of the two step iteration (11a) and (11b) we could just use the obvious four step analog.

Should we worry about the fact that s and g are orthogonal variables but s and g along with $y = (s+g)/2$ and $h = (g-s)/2$ are not? It may be helpful to consider the problem in Fourier space. All the axes may be Fourier transformed to say k_s, k_g, k_y and k_h. Processes like (11a) and (11b) amount to forming residuals by subtracting certain Fourier components from the data. For example, the very classical problem (1) amounts to subtracting a Fourier component $k_g = 0$ and the component $k_s = 0$. In the (k_s, k_g)-plane, these are the coordinate axes crossing at the origin. The origin is a troublesome point in the classical analysis, because it represents a constant function of s and g which could be added to either of the fitting functions, provided it is subtracted from the other. Bringing in the two new lines $k_y = 0$, and $k_h = 0$ would seem to cause no new problems, other than having additional lines crossing at the origin. Bandwidth about these lines would seem to increase the size of the troublesome point to the size of the region of overlap. Here is where it helps to realize that our iterative procedure minimizes a quadratic. The solution is unique except for the fact that we have accidently chosen **T** to be a positive *semi*-definite matrix. We should have taken it strictly positive, which we could easily do by adding a small amount to the diagonal. The problem of principle disappears. In practice we must be prepared for slow convergence where the filters have overlapping passbands.

Alternative Approaches

Papers on geophysical inverse theory and factor analysis frequently make use of eigen-vector expansions, say something like

$$\mathbf{D} = \lambda_1 u_1 v_1 + \lambda_2 u_2 v_2 + \cdots \qquad \textit{where } \lambda_1 \geq \lambda_2 \geq \cdots \tag{16}$$

This is an attractive model when the dimensionality is small enough that things are computable. Since there could easily be 1000 shot points and 100 offsets, computability of (16) would be a problem. An alternative means of introducing more fitting parameters, advocated by Mitrofanov, is the introduction of some kind of a series expansion such as power series. Philosophically, (16) as well as Mitrofanov's approach seem to be akin to doing spectral analysis by including only the strongest sinusoids, whereas the formulation (7) introduces additional degrees of freedom by allowing a controllable bandwidth about each sinusoid.

The advantage of the approach (7) becomes especially clear in the case where a strong noise occurs with a weak signal. Accurate treatment of the noise may be required

before the signal becomes visible. Such accurate treatment may be impossible. Even though the noise may have some well known dip or velocity, nothing is ever perfect and the noise demands some bandwidth in its representation before you can go on to look for smaller things.

Slant Stacks and Vertical Seismic Profiles

The data of a simple vertical seismic profile is generated by a sound source near the top of the bore hole. Receivers distributed at various depths within the hole provide a (z,t)-plane of data. In practice, there will be some departures from the idealized model. The shot point may be somewhat offset from the top of the hole. The vertical positioning of the receivers does not provide exactly uniform spacing in travel time. There are often strong tube waves vertically propagating in the well bore itself. Information may be extracted by modeling the downgoing p-wave, the up going p-wave, or various shear and converted waves. In much of the (z,t)-plane the downgoing waves dominate making it difficult to see the upcoming waves. An interesting aspect of the practical problem is that the depth interval between receivers is often sufficient for the low frequencies, but inadequate for the higher frequencies. More precisely, time aligning the data for the downgoing waves often leaves the upcoming waves spatially aliased. Another interesting aspect of the practical problem is that it is much easier to acquire the data uniformly spaced on the z-axis than to obtain it uniformly spaced in travel time depth. Even if you get it uniformly spaced for the p-waves it probably won't be uniformly spaced for the tube waves and s-waves.

This may be ideal data for the methods of the present paper. The technique would be to first time align on the strongest event, say the downgoing wave. By means of an equation like (11a), decompose the data set into two parts, the downgoing wave and its residual. Then time shift the residual for the next strongest wave and continue likewise. Of course a certain amount of hand crafting effort must go into the issue of choosing the best bandwidth for each dip component.

There is a reason to expect that unlike a slant stack representation, we can hope to be somewhat immune to the problems of truncation and aliasing of strong events. Why do we tackle the strong events first? Constant dip components in the (z,t)-plane show up as straight lines thru the origin in the (k_z,ω)-plane. As mentioned earlier, the origin where such events cross one another is something of a problem. Because the data is *sampled* in physical space, in Fourier space it is *replicated* periodically into "tiles". Ideally, the data is not aliased and energy in the basic tile does not extend out to the periphery of the tile. In reality, data is often aliased in z. Thus we have lines intersecting, not only at the origin, but

also where lines from one tile cross those of a neighboring tile. By selecting strong events first, we effectively assume that *all* of the energy at the ambiguous point belongs in the strong event. Of course this is not strictly true, but it may be a lot better than other simple assumptions. One such simple assumption might be that different components in dip space are orthogonal to one another. Such an assumption might be appropriate in an infinite continuum, but could be quite troublesome for slant stacks of truncated data.

Multiple Reflections

There is an old process for removing marine multiple reflections which usually fails in practice. It is worthwhile reviewing why it fails to see what may be done. The method is this:

First moveout the data by the multiple velocity.
Then form a stack to the zero offset trace.
Then subtract this stack from all the offsets.

You can easily imagine all the things which can go wrong. Even in a perfect world, the individual traces are not identical to the stack. Reflection coefficient is a function of angle. Shot and geophone antennas and ghosts are angle dependent. The NMO process does a differential wavelet stretch. In the real world, the list of possible causes of discrepancy is much longer.

The repair needed to the old multiple removal process is simply to replace "stack" by "low pass filter" over offset. The offset axis is a short one, which means that there is almost as much "end" as there is "middle". The end effects are nicely handled by the the quadratic form indicated by equations (4) to (5).

In the discussion of vertical seismic profiles, there was frequent reference to "straight lines" in both data space and Fourier space. This idea of lines was helpful for discussion purposes, but it is not at all required by the theory. The seismic events could easily be parabolic or hyperbolic. All we require is the ability to time shift to flatten one at a time. We could model pegleg events as well as sea floor events.

Hand Crafting the Bandwidth

Let us return to the issue of the choice of the quadratic form **T**. This could be done many different ways and each would give a different answer. Until now the only aspect of arbitrariness which has been stressed is that a scaling factor in **T** determines the decay

rate in the low pass filter. Computational limitations force us to stay within the realm of narrowly banded matrices, but that still leaves us with a lot of freedom. For example we could as well have penta-diagonal matrices, but then how would we choose the functional form?

My philosophy about choice of weights is based on the idea that the method of least squares seems to abhor concentrating residuals if the constraints will allow them to spread out. In other words, in the absence of information, there is a tendency for residuals to be uniformly distributed. If you have any prior idea, or any learned idea, of how the correct answer **a** should look, then you should design **T** to be inverse to it. In other words, **T** should be inverse to the expectation of $\mathbf{a}\bar{\mathbf{a}}^*$.

No one has ever explained to me what the "inductive" method of scientific inference really is, but I always imagine it is something like this: Use theory to get your first guess of signal bandwidth, that gives a **T**. Using it, do some computation to get a better idea of the behavior of **a**. Some appropriate averaging should now help you bootstrap yourself up to a better **T**.

Computability of multidimensional filters is a problem. There are not very many mathematical forms which will be manageable as is the product form of (12) and (13). Another family of two-dimensional computable weights is suggested by the recursive dip filter paper.

Converted Waves

Theory predicts that we should observe pressure to shear converted reflections at non-zero offset. In practice, the offset often equals or exceeds the depth of drilling targets. So the converted waves should often be big enough to see. Yet interpreters rarely try to see converted reflections, and I don't believe I ever have seen them. The obvious procedure is to form a stack according to the appropriate moveout. This always gives an answer but it is not self evidently correct. In certain geometries converted waves should be as self-evident as peg-leg multiples. But I haven't seen this.

Alternately, one could hope to routinely observe shear velocities emerging from velocity studies, but this doesn't seem to be the case.

Let us begin our search from the presumption that the major problem in finding and identifying converted waves is the presence of ordinary pressure waves. Calling this noise, our first task is to filter this from the three dimensional data set (time, midpoint, and offset). The first step is normal moveout (and perhaps statics and prestack partial migration). Now the data should be approximately a constant function of offset, although it may be almost any function of time and midpoint. As described earlier, we choose some kind of a moveout

bandwidth, low-pass for the p-waves, and subtract them from the data set.

Since we have not stacked the data, we have not reduced its dimensionality from three dimensions to two. This is where a movie machine can be helpful. We do not have a very good idea of the s-wave velocities. Furthermore, our best chance of observation lies at large offset where the abnormal moveout is greatest. So we really do not wish to stack. We hope movie display of the residuals will be the appropriate means of interpretation.

In a flat earth, it may be almost impossible to distinguish converted waves from low velocity multiples on the basis of moveout alone. Note that the reflection point for converted shear waves does not lie under the midpoint. So maybe dip is the key to distinguishing. The best strategy may be to seek some display the residual data movie so that converted waves, if present, can be distinguished from low velocity multiple reflections. This is a topic for future research.

Generalized Statics: Product Separability

Propagation of waves is often modeled as a cascade of linear filters, say for example a near-source, soil-resonance filter, a downward propagation filter, a reflection filter, upward propagation, and finally a near-receiver, soil-resonance filter. Presuming each of these effects to be a pure delay say t_j then the Fourier response is $e^{i\omega t_j}$. So in principle, logarithms convert a problem of cascaded filters to a problem of additive times t_j. If the filters have amplitude scaling, as well as delay, the situation is still additive in the complex logarithm. If the amplitude is a function of frequency, then the amplitude problem fits the "additive, almost-separable" model, but we are left with the puzzle of the phase. In practice, it doesn't work out. The first sign of trouble is that you must find the logarithm of the Fourier transform of the observed waveforms. The imaginary part of the logarithm is an angle, something which has an arbitrary additive multiple of 2π. Any scheme for choosing the multiple of 2π must attempt to achieve continuity in ω or pay the price of a a ragged, spread-out time response. Any attempt to achieve continuity in ω must eventually come to grips with poor signal to noise ratio at both ends of the spectrum, and perhaps at places inbetween. Rather than continue to confront the mathematical problem, it is worthwhile to consider the physical problem. The imaginary part of the logarithm gives you the time t_j. Data may be a superposition of many wave paths each with a different t_j. So the answer to the phase question is really multivalued, and any attempt to force uniqueness is doomed.

We can defer the phase question by taking the *power* spectrum of the data. Since power is positive, the logarithms are additive, and the power spectra may be decomposed easily. But to decompose the data set itself requires that you find some way to invent a

phase spectrum for each filter. Causality offers some help, but does not solve the problem for you because filters with arbitrary delays are also causal. In practice, the minimum phase assumption can be made (Taner) or there may be some physical condition such as no reflections from within a water path (Morley) which provides a filter. My purpose here is to try to point another way through this tangle.

Recall the algorithm implied by (11). We can think of this algorithm in more general terms. Recall that there are many "debubble" algorithms besides the classical minimum phase deconvolution. For example, Wiggins' MED process claims to be able to find a waveform, be it minimum phase or not, which is common to a group of waveforms. Thus we can imagine an iterative process which seeks such a waveform in the (g,t)-plane, then moves on to the (s,t)-plane, the (h,t)-plane, etc, and finally iterates and cycles through all these planes. The MED process cannot determine overall time shifts, but it should be able to determine other non-minimum phase shifting. You achieve a lot without even considering the overall time shifts. Morley's sea-floor consistent dereverberation is an impressive process, but he wouldn't attempt it in shallow water. Anyway, if time shifts are desired, it seems best to seek them after surface reverberations have been compensated.

REFERENCES

Mitrofanov, G.M., *Effective Representation of the Wave Field in Seismic Exploration*, Soviet Geology and Geophysics (Geologiya i Geofizika), Vol. 21, No. 4, pp. 135-144, 1980

Claerbout and Hale, *Recursive Dip Filters* SEP 20 p235