Dr. V. Kratky
National Research Council of Canada
Ottawa, Ontario, Canada, K1A OR6

Reflexive Prediction and Digital Terrain Modelling

Array algebra is well suited for reflexive prediction when used with data sets in a grid pattern.

Introduction

Bjerhammar (1975) formulated an estimation process which he called reflexive prediction. A set of n given values f is related to m fictitious, unknown observations h by optimal prediction, i.e., with minimum variance, by using an *a priori prescribed* covariance function. The unknown observations are indirectly represented, in selected carrier points, by means of computed parameters x used later for any new predictions. The basic relations are

$$f = k^T Q^{-1} h = k^T x \tag{1}$$

and

$$f = KQ^{-1} h = K x \quad , \tag{2}$$

where $K(n,m)$ and $Q(m,m)$ denote matrices with rows represented by covariance vectors k^T.

Abstract: *Two-dimensional prediction is often used in connection with data points distributed over a rectangular grid. If the prediction process is based on a covariance function separable in coordinates, conditions are set for an effective use of Rauhala's array algebra. Its advantages are demonstrated for the 'reflexive' type of prediction implemented with or without least squares filtering. Some effects of this formulation upon digital terrain modelling are discussed.*

Parameters x typical for the carrier point configuration are computed from given f with the use of known K. Bjerhammar presents four versions of the procedure, two of which are of interest to this paper:

- pure prediction without filtering, for $n = m$

$$x = K^{-1} f \quad , \quad h = f \, , \tag{3}$$

- prediction with least squares filtering, for $n > m$

$$f = K x \quad , \quad x = (K^T P K)^{-1} K^T P f \, , \tag{4}$$

where P^{-1} is a covariance matrix of the noise in f.

Grid-Structured Reflexive Prediction

Both formulations can be conveniently applied to grid-structured sets of data points. In this instance, one can achieve a considerable saving in computation time by reducing the size of solution matrices to be inverted, in a modification of formulas based on the concept of array algebra (Rauhala, 1974, 1976; Kratky, 1976). In this formulation data elements are always considered to form full and regular arrays in the appropriate n-dimensional space,

i.e., they are organized in regular grid patterns with no gaps within. A vector can represent only one-dimensional distribution of data and is not used to assemble data of arbitrary patterns from higher order spaces. In connection with reflexive prediction, two additional conditions must be met:

- the noise in $\boldsymbol{f}$ should be random and stationary, i.e. $\boldsymbol{P} = \boldsymbol{I}$,
- the adopted covariance function must be *separable* in coordinates x and y (Woltring, 1977).

Deviating now from Bjerhammar's original notation throughout this paper, in favor of a more frequently used one, the covariance function dependent on the distance d between two points is written as

$$c(d) = c(\Delta x) \cdot c(\Delta y) \quad .$$

Based on the assumption of a square grid with a total of n^2 data points arranged in a vector $\boldsymbol{l}$, the predicted value $\hat{s}$ for an arbitrary point is derived by a formula equivalent to Equation 1, i.e.,

$$\hat{s} = \boldsymbol{c}^{\mathrm{T}} \boldsymbol{Q}^{-1} \boldsymbol{l} = \boldsymbol{c}^{\mathrm{T}} \boldsymbol{g} \quad , \tag{1a}$$

where $\boldsymbol{c}$ is a covariance vector, $\boldsymbol{Q}$ is a covariance matrix and $\boldsymbol{g}$ is a vector of the auxiliary parameters associated with the given configuration of carrier points. Following Bjerhammar's reversal one can consider the values in $\boldsymbol{l}$ as nonexistent, unknown observations, ignore them in Equation 1a and introduce a set of new, *given observations* $\bar{l}$ instead of predictions $\hat{s}$. These new observations described by vector $\bar{\boldsymbol{l}}$ are distributed in a raster of m^2 points in another, denser grid over the same area of carrier points. Since the resulting equation system is now overdetermined, corrections $\boldsymbol{v}$ are associated with observation $\bar{\boldsymbol{l}}$ to make the equations consistent

$$\underset{(m^2,1)}{\bar{\boldsymbol{l}}} + \boldsymbol{v} = \underset{(m^2,1)}{\hat{\boldsymbol{l}}} = \underset{(m^2,n^2)}{\boldsymbol{C}} \quad \underset{(n^2,1)}{\boldsymbol{g}} \quad . \tag{2a}$$

If the elements of $\bar{\boldsymbol{l}}$, $\boldsymbol{l}$, $\boldsymbol{v}$, and $\boldsymbol{g}$ are arranged in two-dimensional arrays $\bar{\boldsymbol{L}}$, $\boldsymbol{L}$, $\boldsymbol{V}$, and $\boldsymbol{G}$, respectively, in accordance with the natural location of individual data points in rows and columns of both grids, Equations 1a and 2a change into a form

$$\underset{(1,1)}{\hat{l}} = \underset{(1,n)}{\boldsymbol{c}_y^{\mathrm{T}}} \quad \underset{(n,n)}{\boldsymbol{G}} \quad \underset{(n,1)}{\boldsymbol{c}_x} \tag{5}$$

and

$$\underset{(m,m)}{\bar{\boldsymbol{L}}} + \boldsymbol{V} = \underset{(m,m)}{\hat{\boldsymbol{L}}} = \underset{(m,n)}{\boldsymbol{C}_y} \quad \underset{(n,n)}{\boldsymbol{G}} \quad \underset{(n,m)}{\boldsymbol{C}_x^{\mathrm{T}}} \tag{6}$$

where $\boldsymbol{c}_x$, $\boldsymbol{c}_y$ are vectors formed from coordinate components of covariance functions, and $\boldsymbol{C}_x$, $\boldsymbol{C}_y$ are corresponding matrices composed of row vectors $\boldsymbol{c}_x^{\mathrm{T}}$, $\boldsymbol{c}_y^{\mathrm{T}}$, respectively.

Covariance functions are very often expressed by a Gaussian function

$$c(d) = \exp(-k\, d^2),$$

and since

$$d^2 = \Delta x^2 + \Delta y^2 \quad ,$$

the factorization of $c(d)$ obviously yields

$$c(d) = c(\Delta x) \cdot c(\Delta y) = \exp(-k_x \Delta x^2) \cdot \exp(-k_y \Delta y^2) \quad .$$

A side benefit of the covariance factorization is that the spread of the Gaussian function can be controlled independently in both basic directions if the grid intervals are different (Kratky, 1975).

The original vectors $\hat{\boldsymbol{l}}$, $\bar{\boldsymbol{l}}$, $\boldsymbol{v}$, and $\boldsymbol{g}$ can be considered as row-wise expansions of matrices $\hat{\boldsymbol{L}}$, $\bar{\boldsymbol{L}}$, $\boldsymbol{V}$, and $\boldsymbol{G}$, respectively, and the bilinear form in Equation 5,

$$\hat{\boldsymbol{l}} = \boldsymbol{c}_y^{\mathrm{T}} \boldsymbol{G}\, \boldsymbol{c}_x = \sum_{i=1}^{n} \sum_{j=1}^{n} c(\Delta y)_i\, g_{ij}\, c(\Delta x)_j \quad , \tag{7}$$

is then equivalent to the original Equation 1a, i.e.,

$$\hat{s} = \boldsymbol{c}^{\mathrm{T}}\, \boldsymbol{g} = \sum_{k=1}^{n^2} c(d)_k\, g_k$$

where $k = n(i-1) + j,\ k = 1, 2, \ldots, n^2,\ i,j = 1, 2, \ldots, n.$

Solution of Parameters and New Prediction

The least squares condition of minimizing the sum of squares of corrections v leads in the array algebra formulation to

$$\boldsymbol{C}_y^{\mathrm{T}}\, \boldsymbol{V}\, \boldsymbol{C}_x = 0$$

which is an equivalent to the well-known condition

$$\boldsymbol{C}^{\mathrm{T}}\, \boldsymbol{v} = 0$$

in the conventional matrix-vector formulation (Kratky, 1976). From Equations 6 one can directly derive the normal equations

$$\boldsymbol{C}_y^{\mathrm{T}} \boldsymbol{C}_y\ \boldsymbol{G}\ \boldsymbol{C}_x^{\mathrm{T}} \boldsymbol{C}_x = \boldsymbol{C}_y^{\mathrm{T}} \bar{\boldsymbol{L}}\, \boldsymbol{C}_x \tag{8}$$

and then solve for the matrix of parameters

$$\underset{(n,n)}{\boldsymbol{G}} = \underset{(n,n)}{(\boldsymbol{C}_y^{\mathrm{T}} \boldsymbol{C}_y)^{-1}}\ \underset{(n,n)}{\boldsymbol{C}_y^{\mathrm{T}}}\ \underset{(n,n)}{\bar{\boldsymbol{L}}}\ \underset{(n,n)}{\boldsymbol{C}_x}\ \underset{(n,n)}{(\boldsymbol{C}_x^{\mathrm{T}} \boldsymbol{C}_x)^{-1}} = \underset{(n,m)}{\boldsymbol{C}_y^{+}}\ \underset{(m,m)}{\bar{\boldsymbol{L}}}\ \underset{(m,n)}{\boldsymbol{C}_x^{+\mathrm{T}}}, \tag{9}$$

where the simpler form is achieved by using symbol $\boldsymbol{C}^{+}$ for the pseudoinverse of a rectangular matrix. Obviously, the conventional solution will yield

$$\underset{(n^2,1)}{\boldsymbol{g}} = \underset{(n^2,n^2)}{(\boldsymbol{C}^{\mathrm{T}}\boldsymbol{C})^{-1}}\ \underset{(n^2,1)}{\boldsymbol{C}^{\mathrm{T}}\bar{\boldsymbol{l}}} = \underset{(n^2,m^2)}{\boldsymbol{C}^{+}}\ \underset{(m^2,1)}{\bar{\boldsymbol{l}}}\ . \tag{9a}$$

The array solution for grids in which the number of x- or y-intervals differs, will involve the following matrix patterns for $\boldsymbol{G}$ and $\boldsymbol{g}$

$$(n_y n_y)^{-1}\ (n_y n_x)\ (n_x n_x)^{-1} \leftrightarrow (n_y n_x n_y n_x)^{-1}\ (n_y n_x, 1)\ .$$

From the computed parameters, predictions z for a single new point and $\mathbf{Z}$ for an array of points can be determined using Equations 5 and 6, respectively, i.e.,

$$z = \boldsymbol{c}_y^{\mathrm{T}}\, \boldsymbol{G}\, \boldsymbol{c}_x\ ,\ \boldsymbol{Z} = \boldsymbol{C}_y\, \boldsymbol{G}\, \boldsymbol{C}_x^{\mathrm{T}}\ . \tag{10}$$

Computational Aspects

Computer time reduction

Comparison of Equations 9 and 9a shows immediately that a single (n^2,n^2) inverse in Equation 9a is replaced by two (n,n) inverses in Equation 9. In fact, the two (n,n) inverses are numerically identical if the number of grid intervals in both directions is the same. Since the number of multiplications and divisions in an (n,n) matrix inversion is proportional to n^3, the solution of $\boldsymbol{G}$ is faster than that of $\boldsymbol{g}$ by a factor of n^3. Not only does this fact represent an enormous saving of computer time, but it also makes feasible solutions which otherwise would be beyond practical means.

When computing new predictions from Equation 10, the number of operations involved can be quite high and it is interesting to make a comparison between the array and conventional solutions. The following products are based on computations for M^2 new points from n^2 parameters

$$\underset{(M,M)}{\boldsymbol{Z}} = \underset{(M,n)}{\boldsymbol{C}_y}\ \underset{(n,n)}{\boldsymbol{G}}\ \underset{(n,M)}{\boldsymbol{C}_x^{\mathrm{T}}}\ ,\quad \underset{(M^2,1)}{\boldsymbol{z}} = \underset{(M^2,n^2)}{\boldsymbol{C}}\ \underset{(n^2,1)}{\boldsymbol{g}}\ .$$

The total of needed multiplications is $Mn(M+n)$ for the array solution and M^2n^2 for the conventional one. The array solution again proves economically superior by a factor of $Mn/(M+n)$.

The most frequently repeated routine in the prediction process is the computation of covariance vectors. Since values of the Gaussian function die off rapidly with the increased distance, the useful bandwidth of the covariance vector is rather limited. Furthermore, covariances are computed only for regularly spaced series of grid points and, thus, a fairly

limited number of values fully determine the covariance vector needed in the computations. For a Gaussian function whose standard deviation σ is equal to the grid interval, the useful bandwidth includes only seven or nine elements. It is obvious that the computation of covariance vectors can then be conveniently replaced by extracting their values from a look-up table prepared for values ranging from 0 to 4.5 σ. A new point is always within half of the interval to the nearest grid line. This distance defines the position of a pointer in the table, and the rest of the values are extracted automatically by further assignments at equal intervals. Experiments showed that the use of tabulated covariances is about twice as fast as the direct computation of covariance vectors.

COMPUTER SPACE ORGANIZATION

Considering the usually high volume of data in a prediction process, it is imperative that the three (n,n) matrices in Equation 9 needed for the solution be computed in an economical way without wasting computer memory. This could be achieved by forming the matrix products $\boldsymbol{C}_y^T\boldsymbol{C}_y$, $\boldsymbol{C}_x^T\boldsymbol{C}_x$ directly from individual covariance vectors by a sequential accumulation of their column-row products

$$\underset{(n,m)}{\boldsymbol{C}_y^T}\ \underset{(m,n)}{\boldsymbol{C}_y} = \sum_{r=1}^{m} \underset{(n,1)}{\boldsymbol{c}_{y_r}}\underset{(1,n)}{\boldsymbol{c}_{y_r}^T}\ , \quad \underset{(n,m)}{\boldsymbol{C}_x^T}\ \underset{(m,n)}{\boldsymbol{C}_x} = \sum_{s=1}^{m} \underset{(n,1)}{\boldsymbol{c}_{x_s}}\ \underset{(1,n)}{\boldsymbol{c}_{x_s}^T}\ . \tag{11a}$$

The $\boldsymbol{C}_y^T\bar{\boldsymbol{L}}\,\boldsymbol{C}_x$ product in Equation 9 is more intricate and should be formed by a double summation of $\bar{l}$-scaled column-row products of covariance vectors

$$\underset{(n,m)}{\boldsymbol{C}_y^T}\ \underset{(m,m)}{\bar{\boldsymbol{L}}}\ \underset{(m,n)}{\boldsymbol{C}_x} = \sum_{r=1}^{m}\sum_{s=1}^{m} \underset{(n,1)}{\boldsymbol{c}_{y_r}}\ \underset{(1,1)}{\bar{l}_{rs}}\ \underset{(1,n)}{\boldsymbol{c}_{x_s}^T}\ . \tag{11b}$$

Only table-extracted values of covariance vectors are used in the products, and the results are properly positioned within matrices of Equation 9. The same procedure also is applied in the computation of new predictions z or $\mathbf{Z}$ using Equations 10.

Figure 1 gives a graphical representation of the operations in a new prediction z of an individual point. Table-extracted values of covariances cover only a minor portion of vectors $\boldsymbol{c}$ and interact with a suitable submatrix $\boldsymbol{S}$ within $\boldsymbol{G}$ to yield the correct prediction $z = \boldsymbol{w}_y^T\boldsymbol{S}\boldsymbol{w}_x = \boldsymbol{c}_y^T\boldsymbol{G}\boldsymbol{c}_x$ with a reduction of the storage space requirement. The same idea is extended into the prediction of a regular, dense array of new points. In Figure 2 matrices $\boldsymbol{C}$ are formed by a diagonal band of values defined by *gliding* vectors $\boldsymbol{w}^T$. They represent a diagonal ridge in otherwise zero-filled matrices. Furthermore, in grid-structured fields the gliding vectors assume only a limited number of sets of values which repeat periodically, depending on the grid intervals for carrier points and for newly defined points. As a result, the need to store the covariance vectors is restricted to *skewed* submatrices $\boldsymbol{W}$, which are invariant and

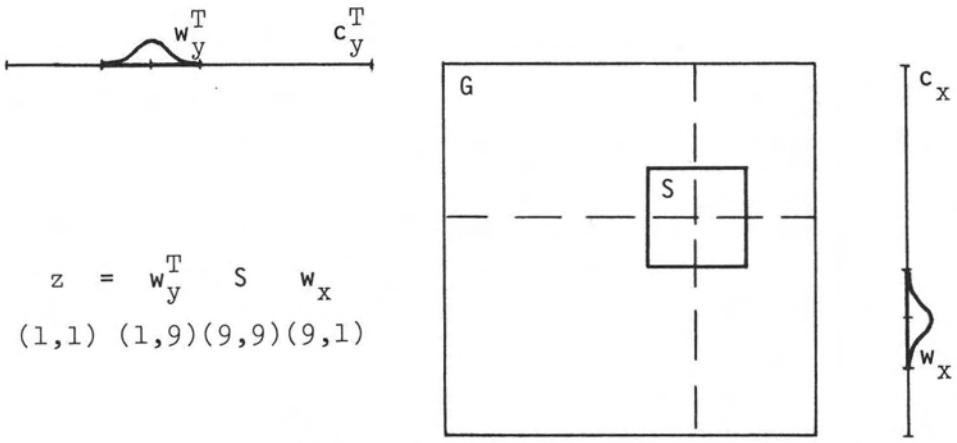

FIG. 1. Prediction for a single point.

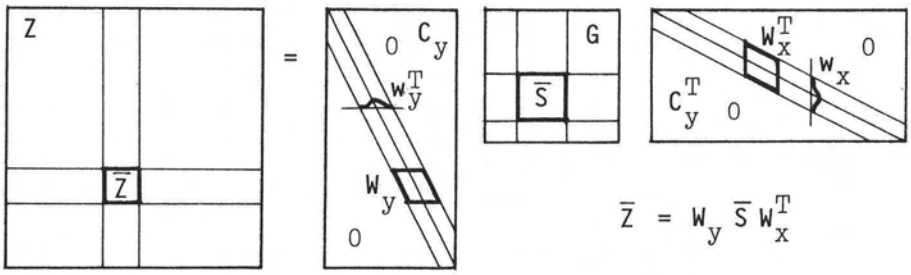

FIG. 2. Prediction for an array of points.

interact with variable submatrices $\bar{S}$ within G to determine individual meshes of the array of new points $\bar{Z} = W_y \bar{S} W_x^T$.

COMPUTATION OF SLOPES

The separation of x,y-components in the covariance function makes it possible to handle not only the basic observed and predicted values, but also their appropriate gradients, e.g., slopes in addition to elevations. The value for z in Equation 10 is expressed as a bilinear form expanded in Equation 7. Slopes ξ, η are defined by partial derivatives dz/dx and dz/dy, respectively. In accordance with the expression in Equation 7 slope ξ is affected only by x-derivatives of $c(\Delta x)_j$

$$c'(\Delta x) = \frac{dc(\Delta x)}{dx} = d(\exp(-k_x \Delta x^2))/dx = 2k_x \Delta x \, c(\Delta x)$$

so that

$$\xi = \frac{dz}{dx} = \sum_{i=1}^{n} \sum_{j=1}^{n} c(\Delta y)_i \, g_{ij} \, c'(\Delta x)_j$$

Defining $\bar{c}_x$ as a vector composed of elements $c'(\Delta x)_j$ and using analogous relations for the y-direction, one arrives at values

$$\xi = c_y^T G \bar{c}_x, \tag{13}$$
$$\eta = \bar{c}_y^T G c_x$$

with vectors $\bar{c}_x$, $\bar{c}_y$ composed of derivatives

$$c'(\Delta x) = 2k_x \Delta x \, c(\Delta x) \quad , \quad c'(\Delta y) = 2k_y \Delta y \, c(\Delta y) \ .$$

These values are easy to derive at the time when covariances $c(\Delta x)$, $c(\Delta y)$ are computed.

With the knowledge of slopes ξ, η it is possible to determine the local direction α of a contour

$$\alpha \simeq \tan^{-1}(-\xi/\eta)$$

and the magnitude of the local maximum slope

$$\rho^2 \simeq \xi^2 + \eta^2 \ .$$

Both these values are important for procedures which reconstruct contour lines from digital terrain models. A single set of parameters G contains full information needed for both the densification of z and for contouring.

MODEL GENERALIZATION

In the solution of Equations 6 the information contained in the original m^2 observation data points was compressed by the least squares procedure into n^2 carrier points represented here by parameters g. One can easily find the corresponding set of unknown observations l in these points. The degree of change introduced by the compression is characterized by corrections v applied to the original observations $\bar{l}$. Obviously, the corrections represent the

effect of the least squares filtering of the original data. Values $\hat{l}$ resulting from this filtering are identical with the predictions which could be derived from parameters g or, indirectly, from pseudo-observations l in the condensed array of carrier points.

The filtering has a smoothing effect on the analytical surface modelled by the prediction and represents certain generalization of its features. With the use of previous formulations parameters associated with the given configuration of carrier points can be directly converted into a different set of parameters typical for another configuration of carrier points. In accordance with Equation 6 the original observations in array $\bar{\boldsymbol{L}}(m,m)$ are also related to parameters $\bar{\boldsymbol{G}}$ in coincident carrier points by

$$\bar{\boldsymbol{L}} = \bar{\boldsymbol{C}}_y \bar{\boldsymbol{G}} \bar{\boldsymbol{C}}_x^{\mathrm{T}} \tag{14}$$

where all matrices have the same dimension (m,m). By substituting this expression for $\bar{\boldsymbol{L}}$ in Equation 9, one obtains

$$\boldsymbol{G} = \boldsymbol{C}_y^{+} \bar{\boldsymbol{C}}_y \bar{\boldsymbol{G}} \bar{\boldsymbol{C}}_x^{\mathrm{T}} \boldsymbol{C}_x^{+\mathrm{T}} = \hat{\boldsymbol{C}}_y \bar{\boldsymbol{G}} \hat{\boldsymbol{C}}_x^{\mathrm{T}} \; . \tag{15}$$

This formula represents a direct compression of a given parameter array $\bar{\boldsymbol{G}}(m,m)$ into a new array $\boldsymbol{G}(n,n)$ simply by multiplication of matrices derived from covariance vectors, i.e., without using any observation data. The compression can be applied in two different ways; either maintaining the same area of the model and enlarging the grid interval or maintaining the interval and reducing the size of the model. The latter method of implementation is typically used for situations in which two or more overlapping digital terrain models are unified and their overlaps eliminated.

Conclusions

The array algebra formulation is well suited for Bjerhammar's reflexive prediction when used with data sets of a grid pattern. By its use, computer time and memory requirements for solutions in large models are drastically reduced. Digital models are analytically defined by parameters which are applied to compute model values, as well as their gradients. It may be advantageous to determine a regular array of parameters and preserve them for further processing even if original data are not available in a grid pattern.

References

Bjerhammar, A. 1975. "Reflexive Prediction," *Vermessung, Photogrammetrie, Kulturtechnik,* 73:3,4, pp. 243-246.

Kratky, V. 1975. "Digital Modeling of Limbs in Orthopaedics," *Photogrammetric Engineering and Remote Sensing,* 41:6, pp. 741-752.

_____ 1976. "Grid-Modified Polynomial Transformation of Satellite Imagery," *Remote Sensing of Environment,* 5:1, pp. 67-74.

Rauhala, U.A. 1974. "Array Algebra with Applications in Photogrammetry and Geodesy," *Fotogrammetriska Meddelanden,* VI:6, p.144.

_____ 1976. "A Review of Array Algebra," Presented Paper to the XIIIth Congress of ISP, Helsinki.

Woltring, H.J. 1977. "Bilinear Approximation by Orthogonal Triangularization," *Photogrammetria,* 33:3, pp. 77-93.

Forthcoming Articles

Ing. Henri Audirac P., Color Infrared Film for Mapping and Multispectral Photointerpretation.

Jon S. Beazley, P.E., Perimeter Survey by Photogrammetric Methods.

John R. Jensen, Ph.D., John E. Estes, and *Larry R. Tinney,* High-Altitude versus Landsat Imagery for Digital Crop Identification.

John F. Kenefick, Donald E. Jennings, Warren Fusselman, L.S., and *Norman Millichamp,* Bridging with Independent Horizontal Control.

A. O. Quinn, P.E., Kenneth E. Reynolds, Memorial Address.

David J. Sauchyn and *Nicholas R. Trench,* Landsat Applied to Landslide Mapping.

John W. Sheldon, In Situ Measurement of Water Transparency.

Raymond D. Spencer, Map Intensification from Small Format Camera Photography.