Polynomial-Like Approximation

Frequently, in machine computations, one may substitute for a function of relatively complicated analytic structure a function of simpler form without appreciable error. Moreover, if the function is given tabularly, a considerable reduction of tabular input data (load on the memory) may be achieved. In what follows, we deal with real-valued functions of continuous real variables $x, y, \cdots$. Computationally, it is usually better to work with variables assuming finitely many discrete values. In the formulas which follow, however, the discrete analogues are apparent, integration to be replaced by summation, etc. We have in mind multivariate approximations by polynomial-like forms, in particular, by functions M of the form

$$(1) \qquad M(x, y) = \sum_{j=1}^{n} \phi_j(x)\psi_j(y),$$

in the bivariate case. Some of the following results are generalizable to functions of more than two variables.

Of special interest is the so-called slide rule form, $f(x) + g(y)$. We shall give formulas for approximating to a given real-valued continuous function, defined over the unit square $0 \leq x, y \leq 1$ to minimize, respectively, each of the following measures of goodness of fit:

(a) $\qquad \int_0^1 \int_0^1 [z(x, y) - f(x) - g(y)]^2 dxdy,$ and

(b) $\qquad \max_{0 \leq x, y \leq 1} |z(x, y) - f(x) - g(y)|.$

A simple variational technique yields for (a):

$$f(x) = \int_0^1 z(x, y)dy + c_0, \quad \text{and}$$

$$g(y) = \int_0^1 z(x, y)dx + c_1, \quad \text{where}$$

c_0 and c_1 are any real numbers such that

$$c_0 + c_1 = - \int_0^1 \int_0^1 z(x, y)dxdy.$$

Recursive formulas for (b) are based heuristically on the fact that given two numbers a and b, one can approximate to them by a single number c; taking c to be the arithmetic mean of a and b minimizes the maximum absolute error. The recursive formulas for (b) are as follows:

$$f_{n+1}(x) = \tfrac{1}{2} \{ \max_y [z(x, y) - g_n(y)] + \min_y [z(x, y) - g_n(y)] \},$$

$$g_n(y) = \tfrac{1}{2} \{ \max_x [z(x, y) - f_n(x)] + \min_x [z(x, y) - f_n(x)] \},$$

and f_0 arbitrary but continuous, $n = 0, 1, \cdots$. DILIBERTO & STRAUS[1] have shown quite generally that the above process converges to a pair of continu-

ous functions (f, g) which has the required property. Included in their discussion is a precise method, discovered independently by the writer, for estimating the value of the minimum. In this connection, the reader is referred to their paper.

We now return to the more general form (1). Let $n > 0$ be a prescribed integer. We assume that z (continuous over the unit square) is not of the form (1) for this particular n. We shall give two sets of formulas for approximating to z by continuous functions of the form (1), which satisfy respectively:

(c) The error vanishes on some rectangular grid of lines $x = x_1, x_2, \cdots, x_n$, $y = y_1, y_2, \cdots, y_n$ ($x_i \neq x_j$ and $y_i \neq y_j$ if $i \neq j$)

(d)
$$\int_0^1 \int_0^1 \left[z(x, y) - \sum_{j=1}^n \phi_j(x)\psi_j(y) \right]^2 dxdy \text{ is minimized.}$$

For (c), functions ϕ_j, ψ_j can be computed successively as follows. Choose a point (x_1, y_1) such that $z(x_1, y_1) \neq 0$ and two constants c_1, c_1' such that $c_1 c_1' = 1/z(x_1, y_1)$ and set

$$\phi_1(x) = c_1 z(x, y_1) \quad \text{and} \quad \psi_1(y) = c_1' z(x_1, y).$$

We compute ϕ_2 and ψ_2 in precisely the same manner, except that z is now replaced by a new function z_2 given by

$$z_2(x, y) = z(x, y) - \phi_1(x)\psi_1(y).$$

Thus, we choose a point (x_2, y_2) such that $z_2(x_2, y_2) \neq 0$, and two constants c_2, c_2' such that $c_2 c_2' = 1/z_2(x_2, y_2)$ and set

$$\phi_2(x) = c_2 z_2(x, y_2) \quad \text{and}$$
$$\psi_2(y) = c_2' z_2(x_2, y).$$

We continue in this manner, with

$$z_{k+1}(x, y) = z_k(x, y) - \phi_k(x)\psi_k(y)$$

until we have the required number of functions. A simple inductive argument shows that condition (c) is satisfied.

That the solution of problem (d), i.e. finding a least squares fit to z of the form (1), bears a direct relationship to the theory of integral operators in Hilbert space was pointed out by G. W. BROWN a few years ago. His observations are essentially these:

Let $\lambda_1, \lambda_2, \cdots, \lambda_n$ be the *largest* n eigenvalues of the symmetric kernel

$$K(x, y) = \int_0^1 z(x, t)z(y, t)dt$$

and $\phi_1, \phi_2, \cdots, \phi_n$ be corresponding eigenfunctions (normalized so that $\int_0^1 \phi_j^2(x)dx = 1$) and set

$$\psi_j(y) = \int_0^1 z(x, y)\phi_j(x)dx, \quad j = 1, 2, \cdots, n,$$

then the ϕ's and ψ's defined in this way form a solution to (d), the ϕ's being mutually orthogonal and the ψ's being mutually orthogonal. The computations can be carried on sequentially. Let λ_1 be the largest eigenvalue of K, ϕ_1 a corresponding normalized eigenfunction and ψ_1 defined as above. Having found ϕ_1 and ψ_1 we form the symmetric kernel, K_1 associated with the residual function $z(x, y) - \phi_1(x)\psi_1(y)$. Then, the largest eigenvalue of K_1 is the next largest eigenvalue of K and we take ϕ_2 to be the corresponding normalized eigenfunction with ψ_2 defined as above. We continue in this manner until we have the required number of functions. Incidentally, for $n = 1$, a variational technique yields the necessary conditions for an extremum:

$$\phi(x) = \int_0^1 z(x, y)\psi(y)dy \Big/ \int_0^1 \psi^2(y)dy,$$

$$\psi(y) = \int_0^1 z(x, y)\phi(x)dx \Big/ \int_0^1 \phi^2(x)dx.$$

These can be used to generate an iterative computation for ϕ and ψ. However, questions of convergence, proper initiating functions to achieve the largest eigenvalue, etc. seem to be difficult. One final remark—the minimum value of $\int_0^1 \int_0^1 [z(x, y) - \sum_{j=1}^n \phi_j(x)\psi_j(y)]^2 dxdy$ is precisely the sum of the remaining eigenvalues of K, i.e. the sum of the eigenvalues minus the sum of the largest n eigenvalues, the former sum being equal to $\int\int z^2 dxdy$.

<div align="right">OLIVER GROSS</div>

The RAND Corporation
Santa Monica, California

[1] S. P. DILIBERTO & E. G. STRAUS, "On the approximation of a function of several variables by the sum of functions of fewer variables," *Pacific Jn. of Math.*, v. 1, p. 195–210, 1951.

A Logarithm Algorithm

The method of calculating logarithms given in this paper is quite unlike anything previously known to the author and seems worth recording because of its mathematical beauty and its adaptability to high-speed computing machines. Although there are well known methods[1] which involve continued fractions, these methods invariably utilize the *analytic* properties of the logarithm *function* and not the *arithmetic* properties of the *individual* logarithm. The first version of this algorithm is based directly upon such arithmetic continued fractions. In a subsequent skeletonized modification, however, continued fractions no longer appear explicitly.

Let $a_0 > a_1 > 1$ be given. To find $\log_{a_0} a_1$ we determine the two sequences

$$a_2, a_3, \cdots$$
$$n_1, n_2, \cdots,$$

where the n's are positive integers, by the relations

(1)
$$a_i^{n_i} < a_{i-1} < a_i^{n_i+1}$$
$$a_{i+1} = a_{i-1}/a_i^{n_i}.$$