TRANSACTIONS OF SOCIETY OF ACTUARIES
1993 VOL. 45

MULTIDIMENSIONAL WHITTAKER-HENDERSON GRADUATION WITH CONSTRAINTS AND MIXED DIFFERENCES

WALTER B. LOWRIE

ABSTRACT

We present methods that allow the systematic graduation of multidimensional data subject to linear constraints. With these constraints the practitioner can impose limitations on the graduation that agree with prior knowledge. Polynomial or polynomial plus exponential models are implemented. We give an example of a small select and ultimate mortality graduation and an algorithm for generating constraints for such an example. The determination of variances is discussed.

1. INTRODUCTION

The Whittaker-Henderson graduation methods are pragmatic in the sense that an assumed loss function is minimized. The solution of this minimization gives the graduated value. Historically, the results have been good and the method has been easy to set up. With advances in computer hardware technology and in software packages in recent years, algorithms have been developed to perform minimizations under rather general conditions, including the recognition of constraints. It is now possible to solve rather large and complicated optimizations at a reasonable cost.

In light of these developments, this paper shows how to extend multidimensional Whittaker-Henderson graduation to use constraints in a practical and relatively easy way. The use of constraints was motivated by the graduation of select and ultimate mortality tables in which mortality rates, for ages greater than 10–12, increase by issue age, duration, and attained age, except for the dip in the 25–29 age range. Of course, constraints can be used in other applications. At present, the practitioner must graduate select and ultimate mortality tables using cut-and-try techniques to implement well-known inequality relationships in a consistent way [38].

The conversion of multidimensional data into a "linear" order is discussed in the section on vectorization. A practical optimization algorithm to perform the constrained minimization is outlined next. The necessary details to implement the algorithm are given in Appendix C. Under the heading of "loss functions," fit and smoothness are presented. As a part of smoothness, separate and mixed differences are defined. In this paper, mixed differencing is done simultaneously along all coordinates rather than along each coordinate separately. (Kellison [24] calls this "functions of more than one variable.") A presentation on constraints follows.

A simple select and ultimate mortality example is given next to show the practitioner how to set up the method. Computer programs are available, in APL, which will then perform the graduation [31]. The author will provide copies of these programs on request. Also, there is a discussion of the calculation of the variances of various types of graduated values. Some comments on the determination of the smoothing constants are also given. This determination is still not entirely resolved and seems to remain largely a retrospective process that is driven by the *uses* of the graduated values (for example, annually renewable term premiums). We show that under reasonable conditions, there exists a unique set of graduated values and that our method of calculating these values is computationally stable.

Although the example and much of the discussion concern annual mortality rates, the method smooths any type of regularly spaced data. Data that are not regularly spaced can be smoothed with this method by specifying a zero weight at any point at which the method is to supply a value [32].

THE METHOD

Multidimensional Whittaker-Henderson graduation minimizes the loss function

$$L(k) = F + kS, \quad k \geq 0$$

for a fixed but arbitrary value of k, where F has been called a measure of "fit" and S a measure of "smoothness." Actually F measures recognition of the data in the sense that F gets larger as we depart from the data. Also, S gets larger as the graduation gets rougher. Later in the paper we propose the idea that S measures the recognition of a model because S gets larger as the graduation departs from that model. Model functions are used as a criterion of smoothness.

In most practical applications, variations in the data occur because of random fluctuations. By imposing constraints, we remove variations in the data that are considered unreasonable (not feasible) such as negative mortality rates. By using constraints, convexity can be preserved. It has been said that "graduation is a process of substituting impressions for facts."*

OPTIMIZATION ALGORITHM

The constrained optimization algorithm we present is based on the direct solution of the Lagrange first-order necessary conditions for optimality [33, p. 425]. These conditions are linear in the case of quadratic programming and are readily solved.

First, the loss function is transformed into an equivalent form

$$f(\mathbf{v}) = \frac{1}{2}\mathbf{v}^T\mathbf{A}\mathbf{v} + \mathbf{c}^T\mathbf{v} \tag{1}$$

where **v** is the vector of graduated values and the matrix **A** and the vector **c** are defined later in the paper. This form is minimized subject to linear constraints that can be expressed as $\mathbf{E}\mathbf{v}\leq\mathbf{b}$, where **E** is a matrix of constraint coefficients and **b** is a vector of constraint bounds.

The constrained minimization is done in two phases. The two-phase approach is used because the factorization of the matrix **A**, which is done in phase 1, can be used in phase 2. Also, no constraints may be violated in phase 1, making phase 2 unnecessary.

Summary of Algorithm

The first phase solves the quadratic problem:

minimize

$$f(\mathbf{v}) = \frac{1}{2}\mathbf{v}^T\mathbf{A}\mathbf{v} + \mathbf{c}^T\mathbf{v}$$

with no constraints. Then the solution, $\mathbf{v}^*$, is tested to determine whether it satisfies the constraints. If it does, we are done. If not, $\mathbf{v}^*$ is modified to satisfy the constraints (an algorithm to generate constraints and achieve feasibility is given in the example). Call this new estimate $\mathbf{v}_0$. Also, note

*BOESEN, M. Private communication to author, 1991.

that all that is required to begin phase 2 is a feasible solution (an example might be the standard mortality table); the procedure recommended for phase 1 is just one way to obtain a feasible solution.

In phase 2, we solve the constrained quadratic problem:

minimize

$$f(\mathbf{v}) = \frac{1}{2}\mathbf{v}^T\mathbf{A}\mathbf{v} + \mathbf{c}^T\mathbf{v}$$

subject to

$$\mathbf{E}\mathbf{v} \leq \mathbf{b}$$

iteratively using an "active set" method. Active set methods keep track of a "working set" of equality constraints and any inequality constraints that are binding at that stage. The details can be found in Appendix C.

Luenberger [33] states that active set methods are almost always used for quadratic programs with inequality constraints and that factorization methods are commonly used for solving large linear systems.

The matrices for practical problems could be so large the efficiencies of the factorization method are necessary. The matrix divide operation built into APL can be used for small problems.

VECTORIZATION

Suppose the data from a study (for example, select and ultimate mortality) occur in a rectangular array (matrix), say:

$$\mathbf{u} = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{bmatrix} = \{u_{\mathbf{x}}\}$$

where $\mathbf{x}$ is the index, for example, $\mathbf{x}=(2, 3)$, of an element in the matrix. The insight proposed by McKay [34] was to rearrange the array into a vector as follows:

$$v_1 = u_{11}$$

$$v_2 = u_{12}$$

$$v_3 = u_{13}$$

$$\begin{aligned} v_4 &= u_{21} \\ v_5 &= u_{22} \\ &\cdot \\ &\cdot \\ &\cdot \\ v_9 &= u_{33} \end{aligned}$$

so that the vector contains the same data as before but in a linear arrangement. Note that the last coordinate is incremented first, then the next to last, and so on. This process is called vectorization [13], or ravel, in the APL programming language. The purpose of the vectorization is to be able to use the results that apply to the single-dimension case. In general, if a collection of data has D dimensions and the number of entries in each dimension is denoted n_1, n_2, ..., n_D, the original *indices* of the elements are $\mathbf{x}=(x_1, x_2, \ldots, x_D)$, where $1 \le x_1 \le n_1$, $1 \le x_2 \le n_2$, ..., $1 \le x_D \le n_D$. Define $P=n_1 n_2 \ldots n_D$. In the previous example, $n_1=3$, $n_2=3$, $P=9$ [30].

There is a function $\phi(\mathbf{x})$ that converts the original indices $\mathbf{x}$ into the new indices i. We say $i=\phi(\mathbf{x})$, or equivalently $v_i=v_{\phi(\mathbf{x})}$. To define the function ϕ, specify the vector

$$\mathbf{m} = (n_2 n_3 \ldots n_D, n_3 \ldots n_D, \ldots, n_{D-1} n_D, n_D, 1);$$

then $\phi(\mathbf{x})=(\mathbf{x}-1)^T\mathbf{m}+1$. The function ϕ has a unique inverse function $\phi^{-1}(i)$, which is analyzed in Appendix E.

Define the symbols:

v_i a graduated value

u_i an ungraduated value

s_i a standard value

w_i a weight associated with the ungraduated value, u_i

w_i' a weight associated with the standard value, s_i.

Note that the weights are non-negative. If any weights w_i are zero, then the corresponding graduated values are calculated. If any weights w_i' are zero, then the corresponding standard values are ignored.

Further define:

$$\mathbf{v} = (v_1 v_2 \ldots v_p)$$

$$\mathbf{u} = (u_1 u_2 \ldots u_p)$$

$$\mathbf{s} = (s_1 s_2 \dots s_p)$$
$$\mathbf{w} = diag(w_i)$$
$$\mathbf{w}' = diag(w_i').$$

LOSS FUNCTION

Fit

The measure of fit is the weighted sum of the squared deviations of the graduated values from the data. Optionally, the distance from a standard table may be recognized. This measure leads to nice mathematics and has proven to give reasonable results. A flaw of this measure is a tendency to overemphasize outliers. This is usually overcome with small weights where the data are sparse.

Define

$$F = (1 - \alpha)F_1 + \alpha F_2; \quad 0 \leq \alpha \leq 1$$

where

$$F_1 = \sum_{x_1=1}^{n_1} \sum_{x_2=1}^{n_2} \cdots \sum_{x_D=1}^{n_D} w_{\mathbf{x}}(u_{\mathbf{x}} - v_{\mathbf{x}})^2$$
$$= \sum_{i=1}^{P} w_i(u_i - v_i)^2 \qquad (2)$$

and

$$F_2 = \sum_{x_1=1}^{n_1} \sum_{x_2=1}^{n_2} \cdots \sum_{x_D=1}^{n_D} w'_{\mathbf{x}}(s_{\mathbf{x}} - v_{\mathbf{x}})^2$$
$$= \sum_{i=1}^{P} w'_i(s_i - v_i)^2 \qquad (3)$$

where $\mathbf{x}=(x_1, x_2, \dots, x_D)$, and $i=\phi(\mathbf{x})$. Note that in each of Equations (2) and (3), the second summation is just a rearrangement of the first summation. The weights are non-negative.

To be able to use results from linear algebra, we change the form of F_1 and F_2. The first measure of fit becomes:

$$F_1 = (\mathbf{v} - \mathbf{u})^T \mathbf{w}(\mathbf{v} - \mathbf{u})$$

by changing the order of summation in Equation (2) and using vectorization. Similarly,

$$F_2 = (\mathbf{v} - \mathbf{s})^T \mathbf{w}'(\mathbf{v} - \mathbf{s}).$$

By increasing α, we can force the graduation closer to a "standard" table in magnitude and shape. A standard table should be made feasible and transformed as $s_i = as_i' + b$, so that the total number of deaths and the average age at death would match the data. In this way the shape and feasibility of the standard table are preserved [35] if $a>0$.

Smoothness

We give two measures of "smoothness." The first uses separate differences along each dimension, and the second uses mixed differences. The final measure of smoothness is a combination of the first two measures. To discuss smoothness, we define a *model* function as one that remains invariant under the graduation procedure and represents the graduator's idea of smoothness. This concept is discussed later.

Shiu [43] points out that mixed differences are necessary to minimize "cross-product" terms in the model function. Shiu's example is:

$$\underset{1}{\Delta^2}\, x_1 x_2 = 0$$

$$\underset{2}{\Delta^2}\, x_1 x_2 = 0$$

but

$$\underset{1}{\Delta}\,\underset{2}{\Delta}\, x_1 x_2 = 1.$$

So, to ensure the model is of the form

$$ax_1 + bx_2 + c,$$

we must incorporate the mixed difference term in the measure of smoothness.

To examine the case in which the smoothness model includes an exponential term (for example, $r_i>0$ for $i=1, 2$), define:

$$g(x_1, x_2) = (1 + r_1)^{x_1}(1 + r_2)^{x_2}$$

and

$$f(x_1, x_2) = a + bx_1 + cx_2 + d(1 + r_1)^{x_1} + e(1 + r_2)^{x_2}.$$

Then, for example:

$$\underset{1}{\Delta}^3 g(x_1 x_2) - r_1 \underset{1}{\Delta}^2 g(x_1, x_2) = 0$$

$$\underset{2}{\Delta}^3 g(x_1, x_2) - r_2 \underset{2}{\Delta}^2 g(x_1, x_2) = 0$$

$$\underset{1}{\Delta}\underset{2}{\Delta} g(x_1, x_2) = r_1 r_2 g(x_1, x_2) \neq 0$$

$$\underset{1}{\Delta}^3 f(x_1, x_2) - r_1 \underset{1}{\Delta}^2 f(x_1, x_2) = 0$$

$$\underset{2}{\Delta}^3 f(x_1, x_2) - r_2 \underset{2}{\Delta}^2 f(x_1, x_2) = 0$$

$$\underset{1}{\Delta}\underset{2}{\Delta} f(x_1, x_2) = 0.$$

The differences along dimensions one and two do not "detect" the cross-product terms, $(1+r_1)^{x_1}(1+r_2)^{x_2}$, but the mixed difference does. The function $f(x_1, x_2)$, above, represents an excellent criterion of smoothness for a two-dimensional problem.

Separate Differences

Here, the differences are applied separately along each dimension. The degrees of differencing $(z_1, z_2, \ldots, z_D)$ may be different for each dimension $(d=1, 2, \ldots, D)$.

The measure of "smoothness" is defined as:

$$\begin{aligned} S_1 = {} & k_1 \sum_{x_2=1}^{n_2} \sum_{x_3=1}^{n_3} \cdots \sum_{x_1=1}^{n_1-z_1} \left(\underset{1}{\Delta}^{z_1} v_{\mathbf{x}} - r_1 \underset{2}{\Delta}^{z_1-1} v_{\mathbf{x}} \right)^2 \\ & + k_2 \sum_{x_1=1}^{n_1} \sum_{x_3=1}^{n_3} \cdots \sum_{x_2=1}^{n_2-z_2} \left(\underset{2}{\Delta}^{z_2} v_{\mathbf{x}} - r_2 \underset{2}{\Delta}^{z_2-1} v_{\mathbf{x}} \right)^2 \\ & \cdot \\ & \cdot \\ & \cdot \\ & + k_D \sum_{x_1=1}^{n_1} \sum_{x_2=1}^{n_2} \cdots \sum_{x_D=1}^{n_D-z_D} \left(\underset{D}{\Delta}^{z_D} v_{\mathbf{x}} - r_D \underset{D}{\Delta}^{z_D-1} v_{\mathbf{x}} \right)^2 \end{aligned} \tag{4}$$

where $\mathbf{x}=(x_1, x_2, \ldots, x_D)$, and

$$\underset{k}{\Delta} u_{\mathbf{x}} = u_{x_1,x_2,\ldots,x_k+1,\ldots,x_D)} - u_{x_1,x_2,\ldots,x_k,\ldots,x_D}.$$

The complicated notation in Equation (4) is necessary because we must keep track of which elements are subject to the given degree of differencing. The differencing must not "spill over" from one row (column, etc.) to the next in the original multidimensional array.

By reversing the order of the summations and doing considerable manipulation, it can be shown that:

$$\begin{aligned} S_1 &= k_1\mathbf{v}^T\mathbf{K}_1^T\mathbf{K}_1\mathbf{v} + k_2\mathbf{v}^T\mathbf{K}_2^T\mathbf{K}_2\mathbf{v} + \ldots + k_D\mathbf{v}^T\mathbf{K}_D^T\mathbf{K}_D\mathbf{v} \\ &= v^T\mathbf{J}v \end{aligned}$$

where

$$\mathbf{J} = \sum_{j=1}^{D} k_j\mathbf{K}_j^T\mathbf{K}_j.$$

In the following discussion $j=1, 2, \ldots, D$ and s and t denote the rows and columns of $\mathbf{K}$ (respectively); that is, $\mathbf{K}=\{k_{st}\}$.

The individual difference matrices $\mathbf{K}_j$ are defined as:

$$\mathbf{K}_j = \mathbf{K}_j^0 - r_j\mathbf{K}_j^1$$

where

$$\mathbf{K}_j^0\mathbf{v} = \begin{bmatrix} \underset{j}{\Delta}^{z_j} v_1 \\ \underset{j}{\Delta}^{z_j} v_2 \\ . \\ . \\ . \\ \underset{j}{\Delta}^{z_j} v_P \end{bmatrix} \quad \text{and} \quad \mathbf{K}_j^1\mathbf{v} = \begin{bmatrix} \underset{j}{\Delta}^{z_j-1} v_1 \\ \underset{j}{\Delta}^{z_j-1} v_2 \\ . \\ . \\ . \\ \underset{j}{\Delta}^{z_j-1} v_P \end{bmatrix}$$

The differences are defined by:

$$\underset{j}{\Delta^{z_j}} v_s = \begin{cases} \sum\limits_{y_j=0}^{z_j} (-1)^{z_j-y_j} \binom{z_j}{y_j} v_i \; ; & x_j \leq n_j - z_j \\ \\ 0 \; ; & x_j > n_j - z_j \end{cases}$$

where $i=\phi(x_1, x_2, \ldots, x_j+y_j, \ldots x_D)$. To define $\mathbf{K}_j^1$, we note that it must have the same all-zero rows as $\mathbf{K}_j^0$. Then the differences:

$$\underset{j}{\Delta^{z_j-1}} v_s = \begin{cases} \sum\limits_{y_j=0}^{z_j-1} (-1)^{z_j-1-y_j} \binom{z_j-1}{y_j} v_i \; ; & x_j \leq n_j - z_j \\ \\ 0 \; ; & x_j > n_j - z_j, \end{cases}$$

where i and s are the same as in the previous definition. A detailed analysis is presented in Appendix A.

Example of Separate Differences

Suppose the data have three rows and four columns so that $n_1=3$, $n_2=4$. Assume the differences are $z_1=2$ and $z_2=2$. Then $P=n_1n_2=12$ and $D=2$. The following table shows how to determine the all-zero rows in the K-matrices.

		All Zeros	
i	(x_1, x_2)	Δ_1^2	Δ_2^2
1	1, 1		
2	1, 2		
3	1, 3		**Yes
4	1, 4		Yes
5	2, 1	*Yes	
6	2, 2	Yes	
7	2, 3	Yes	Yes
8	2, 4	Yes	Yes
9	3, 1	Yes	
10	3, 2	Yes	
11	3, 3	Yes	Yes
12	3, 4	Yes	Yes

*$x_1>n_1-z_1=3-2=1$.
**$x_2>n_2-z_2=4-2=2$.

Values of t (column index) for non-zero entries in the first row are:

Matrix	Values of t
$\mathbf{K}_1^0$	1 5 9
$\mathbf{K}_1^1$	1 5
$\mathbf{K}_2^0$	1 2 3
$\mathbf{K}_2^1$	1 2

All K-matrices are 12×12 in the example shown on the following page.

According to the previous definitions

$$\mathbf{K}_1 = \mathbf{K}_1^0 - r_1\mathbf{K}_1^1 \quad \text{and} \quad \mathbf{K}_2 = \mathbf{K}_2^0 - r_2\mathbf{K}_2^1.$$

Mixed Differences

Historically, the term "mixed differences" [45] refers to applying several degrees of differences along one dimension. This can be accomplished by adding additional difference matrices, as defined in the following paragraphs.

A measure of smoothness that uses mixed differences is:

$$S_2 = k \sum_{x_1=1}^{n_1-z_1} \sum_{x_2=1}^{n_2-z_2} \dots \sum_{x_D=1}^{n_D-z_D} \Bigg(\underset{1}{\Delta}^{z_1}\underset{2}{\Delta}^{z^2} \dots \underset{D}{\Delta}^{z_D} v_{\mathbf{x}}$$

$$- r\underset{1}{\Delta}^{z_1-1}\underset{2}{\Delta}^{z_2-1} \dots \underset{D}{\Delta}^{z_D-1} v_{\mathbf{x}}\Bigg)^2 \qquad (3)$$

where $\mathbf{x}=(x_1, x_2, \dots, x_D)$. In most applications, r will be zero. The only time a non-zero r would be useful is if the graduator wished to introduce the term $r_1r_2(1+r_1)^{x_1}(1+r_2)^{x_2}$ into the model. Then $r=r_1r_2$.

The term in the summation is defined as zero unless $x_1 \leq n_1-z_1$, $x_2 \leq n_2-z_1$, …, *and* $x_D \leq n_D-z_D$. In Appendix E we show that

$$S_2 = k\mathbf{v}^T\mathbf{M}^T\mathbf{M}\mathbf{v}$$

$$\mathbf{K}_1^0 = \left[\begin{array}{cccccccccccc}
1 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 1 \\
 & & & & & & 0 & & & & & \\
 & & & & & & & & & & & \\
 & & & & & & & & & & &
\end{array}\right]
\qquad
\mathbf{K}_2^0 = \left[\begin{array}{cccccccccccc}
1 & -2 & 1 & 0 & & & & & & & & \\
0 & 1 & -2 & 1 & & & & & & & & \\
0 & 0 & 0 & 0 & & & & & 0 & & & \\
0 & 0 & 0 & 0 & & & & & & & & \\
 & & & & 1 & -2 & 1 & 0 & & & & \\
 & & & & 0 & 1 & -2 & 1 & & & & \\
 & & & & 0 & 0 & 0 & 0 & & & & \\
 & & & & 0 & 0 & 0 & 0 & & & & \\
 & & & & & & & & 1 & -2 & 1 & 0 \\
 & & & & 0 & & & & 0 & 1 & -2 & 1 \\
 & & & & & & & & 0 & 0 & 0 & 0 \\
 & & & & & & & & 0 & 0 & 0 & 0
\end{array}\right]$$

$$\mathbf{K}_1^{l} = \left[\begin{array}{cccccccccccccc}
1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
 & & & & & & & 0 & & & & & & \\
 & & & & & & & & & & & & & \\
 & & & & & & & & & & & & &
\end{array}\right]
\qquad
\mathbf{K}_2^{l} = \left[\begin{array}{cccccccccccc}
1 & -1 & 0 & 0 & & & & & & & & \\
0 & 1 & -1 & 0 & & & & & 0 & & & \\
0 & 0 & 0 & 0 & & & & & & & & \\
0 & 0 & 0 & 0 & & & & & & & & \\
 & & & & 1 & -1 & 0 & 0 & & & & \\
 & & & & 0 & 1 & -1 & 0 & & & & \\
 & & & & 0 & 0 & 0 & 0 & & & & \\
 & & & & 0 & 0 & 0 & 0 & & & & \\
 & & & & & & & & 1 & -1 & 0 & 0 \\
 & & & & 0 & & & & 0 & 1 & -1 & 0 \\
 & & & & & & & & 0 & 0 & 0 & 0 \\
 & & & & & & & & 0 & 0 & 0 & 0
\end{array}\right]$$

where

$$\mathbf{Mv} = \begin{bmatrix} \left(\underset{1}{\Delta}^{z_1} \cdots \underset{D}{\Delta}^{z_D} - r\underset{1}{\Delta}^{z_1 - 1} \cdots \underset{D}{\Delta}^{z_D - 1}\right) v_1 \\ \\ \left(\underset{1}{\Delta}^{z_1} \cdots \underset{D}{\Delta}^{z_D} - r\underset{1}{\Delta}^{z_1 - 1} \cdots \underset{D}{\Delta}^{z_D - 1}\right) v_P \end{bmatrix}$$

Example of Mixed Differences

Suppose the data are given on a 5×6 array of points so $\mathbf{u}=\{u_{x_1 x_2}\}$; $x_1=1$, 2, ..., 5 and $x_2=1$, 2, ..., 6. Then $n_1=5$ and $n_2=6$. Also, assume that the degrees of differencing are $z_1=3$, $z_2=2$. The M-matrices have 12 rows and 30 columns, but the all-zero rows are not shown.

$$M^0 = \begin{bmatrix}
-1 & 2 & -1 & 0 & 0 & 0 & 3 & -6 & 3 & 0 & 0 & 0 & -3 & 6 & -3 & 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 2 & -1 & 0 & 0 & 0 & 3 & -6 & 3 & 0 & 0 & 0 & -3 & 6 & -3 & 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 & 3 & -6 & 3 & 0 & 0 & 0 & -3 & 6 & -3 & 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 & 3 & -6 & 3 & 0 & 0 & 0 & -3 & 6 & -3 & 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 & 3 & -6 & 3 & 0 & 0 & 0 & -3 & 6 & -3 & 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 & 3 & -6 & 3 & 0 & 0 & 0 & -3 & 6 & -3 & 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 & 3 & -6 & 3 & 0 & 0 & 0 & -3 & 6 & -3 & 0 & 0 & 0 & 1 & -2 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 & 3 & -6 & 3 & 0 & 0 & 0 & -3 & 6 & -3 & 0 & 0 & 0 & 1 & -2 & 1
\end{bmatrix}$$

$$M^1 = \begin{bmatrix}
-1 & 1 & 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 1 & 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}$$

For notational convenience, $kS=k_1S_1+k_2S_2$. Additional detail is given in Appendix D.

In the following, we refer to the function $F+kS$ as $f(\mathbf{v})$. Then the function to be minimized is

$$6f(\mathbf{v}) = (1 - \alpha)F_1 + \alpha F_2 + kS = \frac{1}{2}\mathbf{v}^T\mathbf{A}\mathbf{v} + \mathbf{c}^T\mathbf{v}$$

where $f(v)$ is a quadratic function of $\mathbf{v}$, and

$$\mathbf{A} = 2[(1 - \alpha)\mathbf{W} + \alpha\mathbf{W}' + k\mathbf{L}]$$

$$\mathbf{c} = -2[(1 - \alpha)\mathbf{Wu} + \alpha\mathbf{W}'\mathbf{s}].$$

We define $k\mathbf{L}=k_1\mathbf{J}+k_2\mathbf{M}$ for notational convenience, if a combination of separate and mixed differences is used.

The proof of the last two equations for the multidimensional case is the same as the proof for the one-dimensional case because of the vectorization [32].

In the absence of constraints or if none of the constraints is active, the value of $\mathbf{v}$ that produces the minimum value of $f(\mathbf{v})$ is $\mathbf{v}=\mathbf{A}^{-1}\mathbf{c}$. In this case, $\mathbf{v}$ is called an interior point.

CONSTRAINTS

Linear constraints take the following form:

$$\begin{gathered}
e_{11}v_1 + e_{12}v_2 + \cdots + e_{1P}v_P \leq b_1 \\
e_{21}v_1 + e_{22}v_2 + \cdots + e_{2P}v_P \leq b_2 \\
\cdot \\
\cdot \\
\cdot \\
e_{N1}v_1 + e_{N2}v_2 + \cdots + e_{NP}v_P \leq b_N
\end{gathered}$$

or

$$\mathbf{Ev} \leq \mathbf{b}$$

One aspect of using constraints that causes problems is the large number of constraints that are needed. Care in setting up the graduation can reduce the number of constraints. Experience with the method should give criteria for discarding many of the constraints that will never become binding because they are satisfied by a large amount. Redundant constraints must be excluded. An anonymous referee made the excellent suggestion that the dual problem may be more readily solved. Then there would be more variables and fewer constraints since the variables and Lagrange multipliers swap roles.

When select and ultimate tables are graduated, the following types of linear constraints can be used to force the graduated values to follow desired patterns.

Duration: $a_1q_{[x]+t} \leq q_{[x]+t+1}+b_1$ or $a_1q_{[x]+t}-q_{[x]+t+1} \leq b_1$

Issue age: $a_2q_{[x]+t} \leq q_{[x-1]+t}+b_2$ or $a_2q_{[x]+t}-q_{[x-1]+t} \leq b_2$

Attained age: $a_3 q_{[x+t]+s} \leq q_{[x+t-1]+s+1} + b_3$ or

$$a_3 q_{[x+t]+s} - q_{[x+t-1]+s+1} \leq b_3$$

Usually for these types of inequalities:

$$a_1 = a_2 = a_3 = 1 \quad \text{and} \quad b_1 = b_2 = b_3 = 0.$$

The duration constraints are redundant if issue age and attained age constraints are imposed and values of the $a_i > 0$ are all equal. The constants, a_i, could be slightly greater than one to impose an exponential-type pattern up the backward diagonals.

AN EXAMPLE OF SELECT AND ULTIMATE MORTALITY GRADUATION

This application shows how to graduate a select and ultimate mortality table imposing constraints based on the inequalities that were mentioned previously. A small example makes the illustration of the method easier to follow.

Data

The data used were female standard issue ordinary mortality rates of nonmedical issues of 1963–77 from the *TSA 1979 Reports* [10]. Issue age groups 10–14, 15–19, 20–24, and 25–29 were used with durations (policy year) of 1, 6, 11, and 16+. The death rates ($1000q_x$) were:

Issue	Actual				Standard			
Age	1	6	11	16+	1	6	11	16+
10–14	0.171	0.220	0.231	0.530	0.274	0.489	0.560	0.574
15–19	0.330	0.472	0.516	0.750	0.456	0.559	0.585	0.732
20–24	0.253	0.368	0.508	0.935	0.500	0.580	0.715	1.183
25–29	0.267	0.342	0.899	1.664	0.548	0.713	1.190	1.937

The data for policy year 16+ represent ultimate mortality with different issue years and observation periods than the select mortality, so they are not quite comparable. For instance, the standard rates for issue ages 20–24, duration 16+ (1.183) should be equal to or greater than the rate for issue ages 25–29, duration 11 (1.190). Standard mortality is the 1965–70 Female Select and Ultimate Table. The standard table should be made feasible and normalized by using a linear transformation so that the sum of deaths agrees with the data.

The degrees of differencing are assumed to be $z_1=z_2=2$, and the dimensions are $n_1=n_2=4$, $(D=2)$. The exponential constants are $r_1=r_2=0$, and the smoothness coefficients are $k_1=k_2=0.1$. The weights are $w_i=1/16$ for all 16 weights, and $\alpha=0$. Mixed differences were not used in the example.

Constraints

To perform the constrained minimization, we must write the constraints in the form: $\mathbf{Ev}\leq\mathbf{b}$, where $\mathbf{v}$ is a vector of graduated values, $\mathbf{E}$ is a matrix of coefficients of the constraints and $\mathbf{b}$ is a column vector of constraint bounds. To find $\mathbf{E}$ and $\mathbf{b}$ for the example, let the matrix of (graduated) values be denoted:

$$\begin{bmatrix} v_{11} & v_{12} & v_{13} & v_{14} \\ v_{21} & v_{22} & v_{23} & v_{24} \\ v_{31} & v_{32} & v_{33} & v_{34} \\ v_{41} & v_{42} & v_{43} & v_{44} \end{bmatrix} = \begin{bmatrix} q_{[12]} & q_{[12]+5} & q_{[12]+10} & q_{[12]+15} \\ q_{[17]} & q_{[17]+5} & q_{[17]+10} & q_{[17]+15} \\ q_{[22]} & q_{[22]+5} & q_{[22]+10} & q_{[22]+15} \\ q_{[27]} & q_{[27]+5} & q_{[27]+10} & q_{[27]+15} \end{bmatrix}.$$

After the matrix of graduated values is vectorized,

$$v_1 = q_{[12]}, \ldots, v_2 = q_{[12]+5}, v_4 = q_{[12]+15}, v_5 = q_{[17]}, \ldots, v_{16} = q_{[27]+15}.$$

The above expressions use notation that is not consistent with the notation in the rest of the paper. This inconsistency is necessary to introduce standard actuarial notation. Also, in other mortality studies age intervals may not be five.

The constraints are generated in the order shown in the table on the following page. The value 0.0001 appearing in the first line of the table was chosen arbitrarily. This is the only method available to require $q_{[12]}>0$. In general, it is easier to deal with $1000q_x$, so the bounds are multiplied by 1000.

We present an algorithm for generating constraints for select and ultimate mortality tables that omits redundant (duration) constraints and provides a useful order. For instance, if $u_{21}\geq u_{11}$ and $u_{12}\geq u_{21}$, then $u_{12}\geq u_{11}$ is redundant. An expression for the minimum number of constraints is

$$2 + (n_1 - 1)(2n_2 - 1)$$

	Constraints																		
Actuarial Notation	Vector Form	Standard Form								E									
$0.0001 \le q_{[12]}$	$0.0001 \le v_1$	$-v_1 \le -0.0001$	−1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$q_{[12]} \le q_{[17]}$	$v_1 \le v_5$	$v_1 - v_5 \le 0$	1	0	0	0	−1	0	0	0	0	0	0	0	0	0	0	0
$q_{[17]} \le q_{[12]+5}$	$v_5 \le v_2$	$v_5 - v_2 \le 0$	0	−1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
$q_{[17]} \le q_{[22]}$	$v_5 \le v_9$	$v_5 - v_9 \le 0$	0	0	0	0	1	0	0	0	−1	0	0	0	0	0	0	0
$q_{[22]} \le q_{[17]+5}$	$v_9 \le v_6$	$v_9 - v_6 \le 0$	0	0	0	0	0	−1	0	0	1	0	0	0	0	0	0	0
$q_{[12]+5} \le q_{[17]+5}$	$v_2 \le v_6$	$v_2 - v_6 \le 0$	0	1	0	0	0	−1	0	0	0	0	0	0	0	0	0	0
$q_{[17]+5} \le q_{[12]+10}$	$v_6 \le v_3$	$v_6 - v_3 \le 0$	0	0	−1	0	0	1	0	0	0	0	0	0	0	0	0	0
$q_{[22]} \le q_{[27]}$	$v_9 \le v_{13}$	$v_9 - v_{13} \le 0$	0	0	0	0	0	0	0	0	1	0	0	0	−1	0	0	0
$q_{[27]} \le q_{[22]+5}$	$v_{13} \le v_{10}$	$v_{13} - v_{10} \le 0$	0	0	0	0	0	0	0	0	0	−1	0	0	1	0	0	0
⋮	⋮	⋮								⋮									
$q_{[22]+10} \le q_{[27]+10}$	$v_{11} \le v_{15}$	$v_{11} - v_{15} \le 0$	0	0	0	0	0	0	0	0	0	0	1	0	0	0	−1	0
$q_{[27]+10} \le q_{[22]+15}$	$v_{15} \le v_{12}$	$v_{15} - v_{12} \le 0$	0	0	0	0	0	0	0	0	0	0	0	−1	0	0	1	0
$q_{[17]+15} \le q_{[22]+15}$	$v_8 \le v_{12}$	$v_8 - v_{12} \le 0$	0	0	0	0	0	0	0	1	0	0	0	−1	0	0	0	0
$q_{[22]+12} \le q_{[27]+15}$	$v_{12} \le v_{16}$	$v_{12} - v_{16} \le 0$	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	−1
$q_{[27]+15} \le 1$	$v_{16} \le 1$	$v_{16} \le 1$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

where n_1 is the number of rows and n_2 is the number of columns. This expression counts the two constraints that require mortality rates to be greater than zero and less than one.

Further, if the solution of phase 1, v^*, violates any constraints, then v^* may be made feasible by raising the appropriate element to just satisfy each violated constraint. This is done in the order that the algorithm generates constraints and starts at the first violated constraint. For instance, if $u_{21} \leq u_{11}$, then set $u_{21} = u_{11}$. At any step if an element is larger than one, it is set equal to one.

The Constraint Algorithm

The constraints generated by the algorithm are in terms of the original indices **x** to illustrate the pattern followed by the algorithm. Of course the indices would then be vectorized.

The algorithm moves up successive backward diagonals starting with the first. The element at each position on the diagonal has to be larger than the element (if any) immediately above it—and larger than the succeeding element (if any) up the backward diagonal.

The algorithm is as follows:

```
Step-Init.      Input Rowmax($n_1$), Colmax($n_2$)
                Set Diag:=2
                Set Col:=1
Step-Up.        If Col>Colmax stop else continue
                Set Prvrow:=Diag−Col
                If Prvrow=0 go to Step-Newdiag else generate the
                  constraint:
                  $u$[Prvrow; Col]≤$u$[Prvrow+1; Col]
Step-Back.      If Colmax<Col+1 go to Step-Newdiag else generate
                  the constraint:
                  $u$[Prvrow+1; Col]≤$u$[Prvrow; Col+1]
                Set Col:=Col+1
                If Colmax≥Col go to Step-Up else stop
Step-Newdiag.   Set Diag:=Diag+1
                Set Col:=max((1+Diag−Rowmax),1)
                go to Step-Up
```

In phase 1 of our method, the graduation is performed ignoring the constraints. Then a test is made to determine whether any constraints are violated. If not, the process is over. If any constraints are violated, we go to phase 2, where the previous solution is adjusted to be feasible. Then an iterative process is used in phase 2 to find the optimal feasible solution.

The following is the result of the optimization:

Issue	Phase 1. Unadjusted				Phase 2a. Adjusted for Feasibility				Phase 2b. Constraint Optimization			
	Policy Year				Policy Year				Policy Year			
Age	1	6	11	16+	1	6	11	16+	1	6	11	16+
10–14	.209	.260	.323	0.418	.209	.260	.366*	0.522*	.169	.244	.348*	0.512*
15–19	.223	.366	.522	0.707	.223	.366	.522	0.713*	.196	.348	.512	0.734
20–24	.200	.439	.713	1.034	.223*	.439	.713	1.034	.196*	.431	.701	1.025
25–29	.161	.507	.937	1.435	.223*	.507	.937	1.435	.196*	.515	.925	1.406

Note: The asterisks indicate the values that were increased to satisfy a constraint.

The smoothing coefficients were chosen relatively small (0.1) compared to the sum of the weights (1) to ensure that the result of the first graduation (phase 1) would not be feasible for the sake of example. This was done to illustrate the method. Convergence in phase 2 was very fast (2 iterations). The result after phase 2 is not smooth enough but satisfies the constraints. The three equal values in the first column of the final result would not be acceptable in practice. Raising the smoothing coefficients and/or using the standard table (α>0) would make the final graduation smoother. The example ran quickly on an AT&T 6300 with 640K RAM.

EXISTENCE AND UNIQUENESS OF SOLUTIONS

In this section we examine conditions under which the solution is unique. It turns out these conditions are easily attained in practice. A by-product of these conditions is that our algorithm will be unconditionally stable numerically because it uses the Choleski factorization [46, p. 153].

For the one-dimensional case with no constraints, it has been shown that a unique solution exists if all the weights are positive, if standard data are used (that is, α>0) or if there are at least z pivotal points (positive weights) in the interpolation case when the model is:

$$v_i = P_{z-2}(i) + k(1 + r)^i$$

or

$$v_i = P_{z-1}(i), \quad \text{and} \quad r = 0$$

where P_{z-2} is a polynomial of degree $z-2$. The proofs of the above are given in Lowrie [32].

In the following development we show that if the one-dimensional result holds for any one row (column, etc.), then a unique solution exists.

To develop the necessary theory, we introduce the concepts of convex sets, strictly convex functions, and positive definite matrices.

A set C is *convex* if the point $\alpha\mathbf{v}_1+(1-\alpha)\mathbf{v}_2$ is in C for every α, $0\leq\alpha\leq 1$ and every $\mathbf{v}_1$ and $\mathbf{v}_2$ in C. Note that the set $C=\{\mathbf{v}:\mathbf{E}\mathbf{v}\leq\mathbf{b}\}$ is convex.

A function $f:\mathbf{R}_n\rightarrow\mathbf{R}_1$ is *strictly convex* on a set C, if for any two points $\mathbf{v}_1$ and $\mathbf{v}_2$ in $C(\mathbf{v}_1\neq\mathbf{v}_2)$ and for any α, $0\leq\alpha\leq 1$,

$$f[\alpha\mathbf{v}_1 + (1 - \alpha)\mathbf{v}_2] < \alpha f(\mathbf{v}_1) + (1 - \alpha)f(\mathbf{v}_2)$$

If a function f has a Hessian matrix (matrix of all partial second derivatives) that is positive definite on all $\mathbf{v}$ on C, then the function is strictly convex on C. The Hessian of the quadratic function

$$f(v) = \frac{1}{2}\mathbf{v}^T\mathbf{A}\mathbf{v} + \mathbf{c}^\mathbf{T}\mathbf{v}$$

is $\mathbf{A}$, which does not depend on $\mathbf{v}$. So if $\mathbf{A}$ is positive definite, then f is strictly convex on $\mathbf{R}_n$.

Define the constraint set $C=\{\mathbf{v}:\mathbf{E}\mathbf{v}\leq\mathbf{b}\}$. We will need the following lemmas:

Lemma 1. Let

$$f(\mathbf{v}) = \frac{1}{2}\mathbf{v}^T\mathbf{A}\mathbf{v} + \mathbf{c}^T\mathbf{v}$$

where $\mathbf{A}$ and $\mathbf{c}$ are defined previously. Then the matrix $\mathbf{A}$ is positive definite if one of the following is satisfied:

(1) The weights $w_\mathbf{x}$ are positive

(2) Standard values are used ($\alpha>0$)

(3) There is a row (column, etc.) for which at least z_p weights are positive, where z_p is the degree of differencing corresponding to that row (column, etc.).

Proof: See Appendix B.

Lemma 2. Let C be a convex set defined by $C=\{\mathbf{v}:\mathbf{Ev}\leq\mathbf{b}\}$. Let $\mathbf{f}:\mathbf{R}_n\rightarrow\mathbf{R}_1$ be a continuous, strictly convex function on C. For the problem: minimize $\mathbf{f}(\mathbf{v})$ subject to $\mathbf{v}\in C$, there exists a unique global minimum of $\mathbf{f}$ on C.

Proof: See Bazaraa and Shetty [1].

Although not necessary mathematically, $C=\{\mathbf{v}:\mathbf{Ev}\leq\mathbf{b}\}$ will be bounded in most practical applications. For instance, if $q_x\leq 1$ for the last age, then all parameter values will be bounded.

Theorem 1. Given the problem:

minimize

$$\mathbf{f}(\mathbf{v}) = \frac{1}{2}\mathbf{v}^T\mathbf{A}\mathbf{v} + \mathbf{c}^T\mathbf{v}$$

subject to

$$\mathbf{v} \in C.$$

If any of the conditions of Lemma 1 are satisfied, then there exists a unique global minimum of $\mathbf{f}(\mathbf{v})$ on C.

Proof: Since $f(\mathbf{v})=\frac{1}{2}\mathbf{v}^T\mathbf{A}\mathbf{v}+\mathbf{c}^T\mathbf{v}$ is a quadratic function, the Hessian (second derivative) of $f(\mathbf{v})$ is $\mathbf{A}$ and $f(\mathbf{v})$ is continuous. If the Hessian of a function $f(\mathbf{v})$ is positive definite on a set C, then $f(\mathbf{v})$ is strictly convex on C. So the conditions of Lemma 2 are satisfied and $f(\mathbf{v})$ has a unique global minimum on C.

An anonymous referee has pointed out the following result: A function f defined on a non-empty convex set X is said to be pseudo-convex at:

$$x_0 \in X \quad \text{if} \quad (x - x_0)\,\nabla f(x_0) \geq 0 \Rightarrow f(x) \geq f(x_0) \quad \text{for all} \quad x \in X.$$

(A differentiable convex function is pseudo-convex.) In particular, every stationary point of a pseudo-convex function is a global minimum on X, and there are no inflection points [1, p. 510]. If the function is strictly convex on X, then a stationary point is the *unique* global minimum on S.

VARIANCES OF GRADUATED VALUES

Since the variances are not supplied automatically in Whittaker-Henderson graduation formulas as they are in the Bayes' formulas, variances appropriate to the data must be calculated. If the data are mortality rates q_x, then the variances are given by

$$Var(q_x) = \frac{q_x(1 - q_x)}{n_x},$$

if the number n_x is known [30, pp. 10, 11, 14, 25], [25]. If exposures were used to calculate the $\{q_x\}_{x=a,a+1,\ldots,w}$, then approximate variances can be calculated by estimating n_x for each age using an average amount of insurance for each age to obtain estimates of the variances. This process may only yield rough approximations to the true variances.

If the force of mortality can be estimated from grouped data, then one technique that has proven successful, for this author, is to graduate the natural logarithms of the forces of mortality. Then the variances are approximately:

$$\frac{(1 - P_i)}{N_i P_i (\ln P_i)^2}$$

where N_i is the number of survivors at the beginning of the $(i+1)^{st}$ interval and

$$P_i = \frac{N_i - D_i}{N_i}.$$

Then the weight for each group is the reciprocal of the corresponding variance. If the reciprocals of the variances are used as weights for each cell in a parametric method, then we have best linear unbiased estimators of the parameters if k is large and there are no constraints [37].

If values of $\hat{P}_i$ are being smoothed, variances are given in London [30] and Elandt-Johnson and Johnson [14, pp. 140, 158]. Also, Klugman [26] shows how to obtain the standard deviation of the present value of a life annuity as estimated from a graduated mortality table.

An anonymous referee has pointed out that the determination of variances becomes complicated if any of the constraints are active.

> For unconstrained Whittaker-Henderson graduations the graduated values are linear functions of the crude rates, so variances are easy to obtain from the variances of the crude rates. When constraints are introduced (and active), there is no longer a simple functional relationship between the graduated values and the crude values so variances are likely to be impossible to obtain directly.

A bootstrapping method could be employed to determine variances but would be very time-consuming. If the active constraints caused only small adjustments in the graduated rates, the variances (calculated ignoring active constraints) could be taken as rough approximations. The author has performed similar simulations to estimate the variance of complicated functions of weighted least squares parameters [37].

FURTHER CONSIDERATIONS

Smoothness

It is difficult to give a satisfying a priori definition of smoothness for Whittaker-Henderson methods except by specifying a penalty function that gets larger as the graduated values get rougher, in some sense. For further study of moduli (measures) of smoothness, see Ditzian and Totik [12]. The measure of smoothness used in this paper is a variation of the type used in that reference. If there is an error in one value of a sequence of observations of an experiment, then large values of differences appear quickly [24]. Thus, minimizing the sum of squares of third differences will tend to "drive out" a single error.

Smoothness is defined in the theory of splines by minimizing an integral of the square of the second derivative of a function and specifying that the function has continuous second-order derivatives. Whittaker-Henderson and moving-weighted-average methods minimize a sum of squares of differences of the graduated values. The common theme is that the smoothness penalty function is zero for some function and increases as the graduated values depart from that function.

Bayesian graduation has an advantage in this area because it is based on a prior distribution that reflects the graduator's knowledge. Some notion of smoothness may be implicit in the prior distribution, however. As previously stated, the Whittaker-Henderson methods will leave a model function such as a low-degree polynomial (plus an exponential function) invariant. This kind of model function has yielded good results in the past. Also, a relatively large value of k will force the graduated values closer to a least-squares fit of a model function to the data. Thus smoothness is defined by saying the model function is smooth. Note that a high-degree polynomial is usually unsuitable for a model because it may vary too much.

Giesecke [16] proposes polynomial regression as a means of measuring smoothness. Again, a low-degree polynomial is implicitly regarded as the example of smoothness. His methods allow Whittaker-Henderson graduation of irregularly spaced data.

A good model function for a three-dimensional problem is:

$$f(x_1, x_2, x_3) = a + bx_1 + cx_2 + dx_3 + e(1 + r_1)^{x_1} + k(1 + r_2)^{x_2} + m(1 + r_3)^{x_3}.$$

Determination of k and r

Giesecke also proposes a method of using the chi-squared test to choose the smoothing constant k iteratively [16, p. 121]. Also, given a value of k, different values of r could be tried until the loss function is minimum consistent with the corresponding chi-squared value. However, Shiu [43] points out that using a chi-square test is not appropriate for a linear compound graduation formula. If the chi-squared test is not used, then various values of r would be used to estimate the minimum of the loss function, given a fixed value of k. Gill et al. [15] point out that smaller values of a (squared) loss function indicate better estimates of the parameters.

Blending

If a large amount of data is to be graduated, the graduation can be done in pieces and the pieces "blended" together. If the blending is done as follows, feasibility is maintained. Suppose, for example, that a graduation is to be performed for issue ages 20–49 with a select period of five years. One way this could be done is to divide the data in five overlapping groups using issue ages 20–29, 25–34, 30–39, 35–44, and 40–49 and perform the five graduations separately.

Then, for example, blended values q^* are obtained as weighted averages of $q^{(1)}$ (issue ages 20–29) and $q^{(2)}$ (issue ages 25–34) as follows:

$$q^*_{[25]+t} = \frac{1}{2}\, q^{(1)}_{[25]+t} + \frac{1}{2}\, q^{(2)}_{[25]+t}$$

$$q^*_{[26]+t} = \frac{1}{2}\, q^{(1)}_{[26]+t} + \frac{1}{2}\, q^{(2)}_{[26]+t}$$

$$\vdots$$

$$q^*_{[29]+t} = \frac{1}{2}\, q^{(1)}_{[29]+t} + \frac{1}{2}\, q^{(2)}_{[29]+t} \qquad \text{for } t = 0, 1, \ldots, 5.$$

The other groups are blended in a similar fashion. Feasibility is maintained by using this process. Trying to perform the graduation in one large group can take far longer than five times that for one smaller group, or it may not work at all because of the size limitations of the computer.

The conditions under which blending would be useful depend on the computer hardware and software available. Blending may be quite useful if an small 8086 machine is used but unnecessary if a software package like MINOS [36] is used on a big mainframe. Other than this general guide, the author has not determined the conditions under which blending is useful. The applications of blending are a subject for future research.

Comments

There are promising Bayesian methods using constraints; see, for example, Carlin [5] and Carlin and Klugman [6]. Also, there are Bayesian methods that will give an interior point solution that may be smoother than a solution that has binding constraints. If binding constraints prove to give "corners" with the method given in this paper, then an increase in the smoothing constant and/or an increase in α (emphasis on a standard table) would "pull" the graduated values away from the binding constraints.

Brockett uses information theoretic methods to deal with multivariate graduation [2], [3].

Difference matrices could be constructed to minimize $\Sigma(\Delta^z - r\Delta^{z-1})^2$ along diagonals. This may be a matter for further research.

The matrices **A** and **E** are invariably sparse. There is considerable technology devoted to large sparse optimization [9].

The computer optimization packages MINOS [36], GINOS [29] and GRG2 [28] will handle very large problems. If these packages are available, blending will probably not be necessary. Also, these packages could be used with the method of Chan, Chan, and Yu [8] with constraints added to the ℓ_p problem $[2 \leq p \leq \infty]$. A linear programming package such as MPSX (IBM) would solve their l_1 and l_∞ problems with linear constraints added [39].

CONCLUSIONS

The method presented in the paper converges rapidly. It can make rough data smoother, make the data conform to known relationships, and interpolate by setting appropriate weights equal to zero. The mathematical conditions to ensure unique solutions are easily attained in practice. The method is suitable for a microcomputer or a mainframe computer. Larger problems can be done on a microcomputer (or mainframe) by breaking the problem up into segments and blending the result. If similar size problems are done repeatedly, the difference matrix **L** and the constraint matrices **E** and the constraint bound vector **b** can all be stored and used for each problem without recalculation. In the past, the graduation of select and ultimate tables required much judgment and skill [35, p. 50] [30]. The method in this paper is systematic and should reduce or eliminate the necessity for cut-and-try techniques. The method can be applied to a wide variety of one-dimensional or multidimensional problems.

ACKNOWLEDGMENTS

I thank Peter A. Johnson and Bradley P. Carlin for their valuable comments. Also, the anonymous referees have made a considerable positive contribution to this paper. I also want to thank the typist, Jane P. O'Connell, for her dedication and careful work.

REFERENCES

1. BAZARAA, M. S., AND SHETTY, C. M. *Nonlinear Programming*. New York: John Wiley & Sons, 1979.

2. Brockett, P. L., et al. "Information Theoretic Multivariate Graduation," *Scandinavian Actuarial Journal* 2 (1991): 144–53.
3. Brockett, P. L. "Information Theoretic Approach to Actuarial Science: A Unification and Extension of Relevant Theory and Applications," *TSA* XLIII (1991): 73–135.
4. Camp, K. "New Possibilities in Graduation," *TSA* VII (1955): 6–30.
5. Carlin, B. P. "A Simple Monte Carlo Approach to Bayesian Graduation," *TSA* XLIV (1992): 55–76.
6. Carlin, B. P., and Klugman, S. A. "Hierarchical Bayesian Whittaker Graduation," *Scandanavian Actuarial Journal*, in press.
7. Chan, L. K., and Panjer, H. H. "A Statistical Approach to Graduation by Mathematical Formula," *ARCH* 1979.2: 27–60.
8. Chan, F. Y., Chan, L. K., and Yu, M. H. "A Generalization of Whittaker-Henderson Graduation," *TSA* XXXVI (1984): 183–255.
9. Coleman, T. F. *Large Sparse Numerical Optimization*. Lecture Notes in Computer Science. New York: Springer-Verlag, 1984.
10. Committee on Mortality under Ordinary Insurances and Annuities. "Mortality under Standard Ordinary Insurance Issues between 1977 and 1978 Anniversaries," *TSA 1979 Reports* (1980): 7–50.
11. Craighead, S. "Bayesian Graduation," *ARCH* 1991.1: 301–320.
12. Ditzian, Z., and Totik, V. *Moduli of Smoothness*. Springer Series in Computational Mathematics. New York: Springer-Verlag, 1987.
13. Dhrymes, P. J. *Mathematics for Econometrics*. Ed. II. New York: Springer-Verlag, 1984.
14. Elandt-Johnson, R. C., and Johnson N. L. *Survival Models and Data Analysis*. New York: John Wiley & Sons, 1980.
15. Gill, P. E., Murray, W., and Wright, M. H. *Practical Optimization*. New York: Academic Press, 1981.
16. Giesecke, L. "Piecewise Regression as a Measure of Smoothness in Whittaker Graduations," *ARCH* 1984.2: 115–28.
17. Greville, T. N. E. "A Fortran Program for Generalized Whittaker-Henderson Graduation," *ARCH* 1982.1: 385–401.
18. Greville, T. N. E. Discussion of Kimeldorf, G. S., and Jones, D. A., "Bayesian Graduation," *TSA* XIX (1967): 117–20.
19. Hibbs, A. R., and Walford, R. L. "A Mathematical Model of Physiological Processes and Its Application to the Study of Ageing," *Mechanisms of Ageing and Development* 50 (1989): 193–214.
20. Hickman, J. C., and Miller, R. B. "Notes on Bayesian Graduation," *TSA* XXIX (1977): 7–49.
21. Hoem, J. M. "A Contribution to the Statistical Theory of Linear Graduation," *Insurance: Mathematics and Economics* 3 (1984): 1–17.

22. JENNINGS, A. *Matrix Computation for Engineers and Scientists*. New York: John Wiley & Sons, 1977.
23. KABELE, T. G. Discussion of HICKMAN, J. C., AND MILLER, R. B., "Notes on Bayesian Graduation," *TSA* XXIX (1977): 25–42.
24. KELLISON, W. G. *Fundamentals of Numerical Analysis*, 77. Homewood, Ill.: Richard D. Irwin, 1975.
25. KLUGMAN, S. "The Standard Deviation of the Present Value of a Life Annuity as Estimated from a Graduated Table," *ARCH* 1978.2.
26. KLUGMAN, S. "On the Variance and Mean Square Error of Decrement Estimators," *TSA* XXXIII (1981): 301–312.
27. KNORR, F. E. "Multidimensional Whittaker-Henderson Graduation," *TSA* XXXVI (1984): 213–55.
28. LASDON, L. S., WAREN, A. D., AND RATNER, M. W. "GRG2." University of Texas-Austin, School of Business Administration, 1984.
29. LIEMAN, J., ET AL. *Modeling and Optimization with GINOS*. Palo Alto, Calif.: The Scientific Press, 1990.
30. LONDON, R. L. *Graduation—The Revision of Estimates*. Winsted, Conn.: Actex, 1985.
31. LOWRIE, W. B. "A Generalization of Multidimensional Whittaker-Henderson Graduation," *ARCH* 1987.1: 93–144.
32. LOWRIE, W. B. "An Extension of the Whittaker-Henderson Method of Graduation," *TSA* XXXIV (1982): 329–72.
33. LUENBERGER, D. G. *Linear and Nonlinear Programming*. Reading, Mass.: Addison Wesley, 1984.
34. MCKAY, S. F., AND WILKIN, J. C. *Appendix to Experience of Disabled-Worker Benefits under OASDI*. Baltimore, Md.: Social Security Administration, 1975.
35. MILLER, M. D. *Elements of Graduation*. New York: The Actuarial Society of America and the American Institute of Actuaries, 1946.
36. MURTAGH, B. A., AND SAUNDERS, M. A. *MINOS*. Stanford University, Department of Operations Research, 1977.
37. NEAFSEY, P. J., AND LOWRIE, W. B. "A Gompertz Age-Specific Mortality Rate Model of Toxicity from Short-Term Whole Body Exposure to Neutrons in Rats," *Radiation Research* 133 (1991): 234–44.
38. PANJER, H. H., AND RUSSO, G. "A Parametric Graduation of Canadian Individual Mortality Experience 1982–1988 (Revised)." Waterloo, Ont.: Institute of Insurance and Pension Research, 1991.
39. SCHUETTE, D. R. "A Linear Programming Approach to Graduation," *TSA* XXX (1978): 407–446.
40. SEAL, H. L. "Tests of a Mortality Table Graduation," *Journal of the Institute of Actuaries* LXXI (1941): 5–67.
41. SHIU, E. S. W. Discussion of LOWRIE, W. B., "An Extension of the Whittaker-Henderson Method of Graduation," *TSA* XXXIV (1982): 362–68.

42. SHIU, E. S. W. Discussion of CHAN, F. T., CHAN, L. K., AND YU, M. H. "A Generalization of Whittaker-Henderson Graduation," *TSA* XXXVI (1984): 205–208.
43. SHIU, E. S. W. Discussion of KNORR, F., "Multidimensional Whittaker-Henderson Graduation," *TSA* XXXVI (1984): 244–47.
44. SHIU, E. S. W. "Matrix Whittaker-Henderson Graduation Formulas," *ARCH* 1 (1977).
45. SPOERL, C. A. "The Whittaker-Henderson Graduation Formula A—The Mixed Difference Case," *TASA* XLII (1941): 292–313.
46. STEWART, G. W. *Introduction to Matrix Computations*, 153. New York: Academic Press, 1973.
47. TAYLOR, G. C. "The Chi-Square Test of a Graduation by Linear-Compound Formula," *Bulletin de l'Association Royale des Actuaires Belges* LXXI (1976): 26–42.
48. TENENBEIN, A., AND VANDERHOOF, I. "New Mathematical Laws of Select and Ultimate Mortality," *TSA* XXXII (1980): 119–84.
49. WANG, T. C. "Some Comments on the Linear Programming Approach to Graduation," *ARCH* 1988.1: 199–233.
50. WITTEN, M. "A Return to Time, Cells, Systems and Aging: V. Further Thoughts on Gompertzian Survival Dynamics—The Geriatric Years," *Mechanisms of Ageing and Development* 46 (1988): 175–200.

APPENDIX A

Additional details on the construction of difference matrices are as follows:

(1) Determine the rows that are all-zero in $\mathbf{K}_j^0$ and $\mathbf{K}_j^1$ for $j=1, 2, \ldots, D$. This is done by calculating

$$(x_1, x_2, \ldots, x_D) = \phi^{-1}(s)$$

for all indices $s=1, 2, \ldots, P$ $(P=n_1 n_2 \ldots n_D)$. For a fixed but arbitrary value of j, the zero rows are those for which $x_j > n_j - z_j$.

(2) Determine the values of the column index t in the *first row* of K_j^0 that contain non-zero entries. These columns are determined using the equation $t=1+y_j m_j$; $y_j=0, 1, \ldots, z_j$. (There is one less value of t for the matrix $\mathbf{K}_j^1$ than for $\mathbf{K}_j^0$.)

(3) Determine the appropriate entries for the first row of the matrix:

$$k_j^0 = \{k_{st}^0\}$$

where

$$k_{st}^0 = \begin{cases} (-1)^{z_j - y_j}, \dbinom{z_j}{y_j} & \text{using the values of } t \text{ from step 2} \\ 0, & \text{otherwise.} \end{cases}$$

(4) Determine the remaining rows of K_j^0. Each subsequent row is determined by shifting the previous row one position to the right.

To calculate the entries in K_j^1, repeat steps (2) through (4), with z_j replaced by $z_j - 1$, using the same all-zero rows.

APPENDIX B

Lemma 1. Let $f(\mathbf{v}) = \frac{1}{2}\ \mathbf{vAv} + \mathbf{c}^T\mathbf{v}$ where **A** and **v** are defined previously. Then the matrix **A** is positive definite if one of the following is satisfied:

(1) All of the weights $w_{x_1 x_2 \ldots x_D} > 0$

(2) Given $p = 1, 2, \ldots, D$; there are at least z_p positive weights on a "p-row." (For a matrix, a "1-row" is a column since the first coordinate is indexed.)

Proof: To show that A is positive definite, we must show that $\mathbf{a}^T\mathbf{A}\mathbf{a} > 0$ for all $\mathbf{a} \in \mathbf{R}_p$, $\mathbf{a} \neq 0$.

Case 1. This is shown in Chan, Chan and Yu [8].

Case 2. Finally examine the case in which some of the weights are 0. To do this, determine

$$\begin{aligned} \mathbf{a}^T\mathbf{A}\mathbf{a} &= (1 - \alpha)\mathbf{a}^T\mathbf{W}\mathbf{a} + \alpha\mathbf{a}^{\mathrm{T}}\mathbf{W}'\mathbf{a} + \mathbf{a}^T\mathbf{J}\mathbf{a} \\ &= (1 - \alpha)\sum_{i=1}^{P} w_i a_i^2 + \alpha \sum_{i=1}^{P} w_i a_i^2 + \sum_{j=1}^{D} k_j a_j^T \mathbf{K}_j^T \mathbf{K}_j a_j \\ &= (1 - \alpha)\sum_{i=1}^{P} w_i a_i^2 + \alpha \sum_{i=1}^{P} w_i' a_j^2 \end{aligned}$$

$$+ k_1 \sum_{i=1}^{p} \left(\underset{1}{\Delta}^{z_1} a_j - r_1 \underset{1}{\Delta}^{z_1 - 1} a_i \right)^2$$

$$\cdot$$
$$\cdot$$
$$\cdot$$

$$+ k_D \sum_{i=1}^{p} \left(\underset{D}{\Delta}^{z_D} a_i - r_D \Delta^{z_D - 1} a_i \right)^2 .$$

Let $b_{x_1, x_2, \ldots, x_D} = a_i$, where $i = \phi(x_1, x_2, \ldots, x_D)$, then

$$\mathbf{a}^T \mathbf{A} \mathbf{a} = (1 - \alpha) \sum_{x_1=1}^{n_1} \sum_{x_2=1}^{n_2} \cdots \sum_{x_D=1}^{n_D} w_{x_1, x_2, \ldots, x_D} \, b_{x_1, x_2, \ldots, x_D} \qquad \text{(B-1-a)}$$

$$+ \alpha \sum_{x_1=1}^{n_1} \sum_{x_2=1}^{n_2} \cdots \sum_{x_D=1}^{n_D} w'_{x_1, x_2, \ldots, x_D} \, b^2_{x_1, x_2, \ldots, x_D} \qquad \text{(B-1-b)}$$

$$+ k_1 \sum_{x_1=1}^{n_1} \sum_{x_2=1}^{n_2} \cdots \sum_{x_D=1}^{n_D}$$

$$\left(\underset{2}{\Delta}^{z_i} b_{x_1, x_2, \ldots, x_D} - r_1 \underset{1}{\Delta}^{z_1 - 1} b_{x_1, x_2, \ldots, x_D} \right)^2 \qquad \text{(B-1-c)}$$

$$+ k_2 \sum_{x_1=1}^{n_1} \sum_{x_2=1}^{n_2} \cdots \sum_{x_D=1}^{n_D}$$

$$\left(\underset{2}{\Delta}^{z_2} b_{x_1}, x_2, \ldots, x_D - r_2 \underset{2}{\Delta}^{z_1 - 1} b_{x_1, x_2, \ldots, x_D} \right)^2 \qquad \text{(B-1-d)}$$

$$\cdot$$
$$\cdot$$
$$\cdot$$

$$+ k_D \sum_{x_1=1}^{n_1} \sum_{x_2=1}^{n_2} \cdots \sum_{x_D=1}^{n_D}$$

$$\left(\underset{D}{\Delta}^{z_D} b_{x_1, x_2, \ldots, x_D} - r_D \underset{D}{\Delta}^{z_D - 1} b_{x_1, x_2, \ldots, x_D} \right)^2 . \qquad \text{(B-1-w)}$$

For example, choose $p=D$, the sum (B-1-w) will be exactly zero if each of the rows of $\mathbf{U}=\{b_{x_1,x_2,\ldots,x_D}\}$ is determined by the functions

$$b(x_D) = P_{z_D-2}(x_D) + k(1 + r_D)^{x_D} \quad \text{or} \quad b(x_D) = P_{z_D-1} \quad \text{and} \quad r_D = 0.$$

These functions can have, at most, z_D-1 zeros (see Lowrie [32, p. 346]. Consequently, if there are z_p (or more) positive weights in a "p-row," then the sum (B-1-a) is positive, and **A** is positive definite. This argument also applies to all other dimensions.

APPENDIX C

We give additional details to help implement the proposed constrained optimization method. We follow Luenberger [33, p. 423]. The method involves solving the Lagrange first-order necessary conditions. This method is particularly recommended for quadratic programs in the positive definite case.

The constraints that were violated in phase 1 may give rise to the *working set* w_0 of active constraint indices for the second phase if $\mathbf{v}_0$ is calculated by modifying $\mathbf{v}^*$ to be feasible. The "active set" method uses only the currently binding (active) constraints at each iteration.

Phase 1

It is known that the minimum of the quadratic function:

$$f(\mathbf{v}) = \frac{1}{2}\mathbf{v}^T\mathbf{A}\mathbf{v} + c^T\mathbf{v}$$

with no constraints is the unique solution $\mathbf{v}^*$ of $\mathbf{A}\mathbf{v}=-\mathbf{c}$ if **A** is positive definite. For small problems, $\mathbf{v}^*=-\mathbf{A}^{-1}\mathbf{c}$. For larger problems, it is more efficient [33] to solve for $\mathbf{v}^*$ by factoring **A** into $A=LDL^T$ where **L** is lower triangular and **D** is a diagonal matrix with positive elements on the diagonal. Then $\mathbf{v}^*$ is found by a double process of forward substitution and back substitution as follows: $\mathbf{LDL}^T\mathbf{v}=-c$. Let $\mathbf{y}=\mathbf{DL}^T\mathbf{v}$; then $\mathbf{Ly}=-\mathbf{c}$ is solved for $\mathbf{y}^*$ by forward substitution. Finally, we find $\mathbf{v}^*$ from $\mathbf{L}^T\mathbf{v}=\mathbf{D}^{-1}\mathbf{y}$ by back substitution.

Feasibility of v_0

The algorithm given to generate constraints can be used to define the order in which to adjust the values of $\mathbf{v}^*$ for feasibility in a select and ultimate graduation. This process is accomplished by searching through all constraints, beginning at the first constraint that is violated, in the order defined by the algorithm. At each succeeding step, *increase* the value of the variable with the negative coefficient so that the constraint is just satisfied or decrease the value to one, whichever is appropriate. Other types of graduations could use a similar algorithm to adjust for feasibility. Any feasible value can be used to begin phase 2, however.

Phase 2

In phase 2, we use $\mathbf{v}_0$ as the initial estimate of the solution of the problem: minimize $f(\mathbf{v})$ subject to $\mathbf{Ev} \leq \mathbf{b}$. As mentioned above, we use a method of solving the Lagrange necessary conditions, combined with an active set method [33]. The Lagrange method is indicated if $\mathbf{A}$ is positive definite.

Consider the equality constrained problem:

minimize

$$f(\mathbf{v}) = \frac{1}{2}\mathbf{v}^T\mathbf{A}\mathbf{v} + \mathbf{c}^T\mathbf{v}$$

subject to

$$\mathbf{Fx} = \mathbf{b} \tag{C-1}$$

where $\mathbf{F}$ is the matrix whose rows represent the currently *active* constraints.

The Phase 2 Algorithm

Start with a feasible point $\mathbf{v}_0$ and a working set, $\mathbf{w}_0$, of active constraint indices. Set $j=0$.

Step 1. Solve the equality constrained program (C-4). If $\mathbf{d}_j=\mathbf{0}$, perform step 3; otherwise perform step 2.

Step 2. Set $\mathbf{v}_{j+1}=\mathbf{v}_j+\alpha_j\mathbf{d}_j$, where α_j is defined by (C-5). If $\alpha<1$, adjoin the minimizing index in (C-5) to $\mathbf{W}_j$ to form $\mathbf{W}_{j+1}$. Set $j=j+1$ and return to step 1.

Step 3. Compute Lagrange multipliers λ, then let $\lambda_q=\min \lambda_i$. The minimum is taken over all indexes in the working set $\mathbf{W}_j$. If $\lambda_q \leq 0$, stop; v_j is optimal. Otherwise, drop q from $\mathbf{W}_j$ to define $\mathbf{W}_{j+1}$. Set $j=j+1$ and return to step 1.

The Lagrange necessary conditions for the existence of a solution to this problem are

$$\mathbf{A}\mathbf{v} + \mathbf{F}^T\lambda + \mathbf{c} = 0$$
$$\mathbf{F}\mathbf{v} - \mathbf{b} = 0 \tag{C-2}$$

where the matrix

$$\begin{bmatrix} \mathbf{A} & \mathbf{F}^T \\ \mathbf{F} & \mathbf{0} \end{bmatrix}$$

is nonsingular if **A** is positive definite and λ is the vector of Lagrange multipliers.

The equality constrained problem can be solved by solving the linear system:

$$\begin{bmatrix} \mathbf{A} & \mathbf{F}^T \\ \mathbf{F} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -\mathbf{c} \\ \mathbf{b} \end{bmatrix}.$$

At iteration j, a point $\mathbf{v}_j$ is given that is feasible for all constraints and satisfies all the equality constraints of the current working set of indices $\mathbf{W}_j$ (that is, $\mathbf{F}\mathbf{v}_j = \mathbf{b}$). The quadratic program (C-4) corresponding to the working set $\mathbf{W}_j$ is then defined by translating formula (C-1) to the point $\mathbf{v}_j$. This is done by letting $\mathbf{v} = \mathbf{v}_j + \mathbf{d}_j$ in formula (C-1), which results in the program:

minimize

$$\frac{1}{2}\mathbf{d}_j^T\mathbf{A}\mathbf{d}_j + \mathbf{g}_j^T\mathbf{d}_j$$

subject to

$$\mathbf{f}_i^T\mathbf{d}_j = \mathbf{0}, \quad i \in \mathbf{w}_j$$

where

$$\mathbf{g}_j = \mathbf{c} + \mathbf{A}\mathbf{v}_j. \tag{C-4}$$

Then the move in the improving direction is $\mathbf{v}_{j+1} = \mathbf{v}_j + \alpha_j d_j$, where

$$\alpha_j = \text{minimum}\left\{1, \frac{b_i - \mathbf{e}_i^T\mathbf{v}_j}{\mathbf{e}_i^T\mathbf{d}_j}\right\}$$

$$\mathbf{e}_i^T\mathbf{d}_j > 0. \tag{C-5}$$

This minimum is taken over all inactive constraints e_i (outside the working set).

Solution of the Lagrange Equations

The solution of problem (C-4) can be put in terms of solving:

$$\begin{bmatrix} \mathbf{A} & \mathbf{F}^T \\ \mathbf{F} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{d} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -\mathbf{g}_j \\ \mathbf{0} \end{bmatrix}; \mathbf{g}_j = \mathbf{c} + \mathbf{A}\mathbf{v}_j \tag{C-6}$$

for **d** and **λ**. The factorization of **A** was found in phase 1. It can be shown that $\mathbf{A}=\mathbf{LDL}^T$, when **A** positive definite. Using this fact:

$$\begin{bmatrix} \mathbf{A} & \mathbf{F}^T \\ \mathbf{F} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{L} & \mathbf{0} \\ \mathbf{X} & \mathbf{L}^1 \end{bmatrix} \begin{bmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}^1 \end{bmatrix} \begin{bmatrix} \mathbf{L}^T & \mathbf{X}^T \\ \mathbf{0} & \mathbf{L}^{1T} \end{bmatrix}$$

where

$$\mathbf{F}^T = \mathbf{LDX}^T$$

and

$$\mathbf{L}^1(-\mathbf{D}^1)\mathbf{L}^{1T} = \mathbf{X}^T\mathbf{DX}.$$

The product **LD** is lower triangular, so **X** can be found by forward substitution since **L** and **D** have already been computed in phase 1. It can be shown that **X** is of full rank if **F** is of full rank. If **X** is of full rank, then $\mathbf{X}^T\mathbf{DX}$ is positive definite [13, p. 71]. Then $\mathbf{L}^1$ and $-\mathbf{D}^1$ are found by factorization.

According to Luenberger, **X** and $\mathbf{L}^1$ can be updated as a constraint is added or removed from the working set. This feature was not used in the author's APL programs, but can yield efficiency for larger problems. Numerical accuracy deteriorates periodically and the process must then be restarted.

APPENDIX D

ADDITIONAL DETAILS ON THE MIXED DIFFERENCE MATRICES

The mixed difference operator M of degree $z_1, z_2, \ldots, z_D$ applied to $v_{x_1x_2,\ldots,x_D}$ can be expressed as:

$$\underset{1}{\Delta}^{z_1}\underset{2}{\Delta}^{z_2}\ldots\underset{D}{\Delta}^{z_D} v_{x_1x_2\ldots x_D} = \left(\underset{1}{E}-1\right)^{z_1}\left(\underset{2}{E}-1\right)^{z_2}\ldots\left(\underset{D}{E}-1\right)^{z_D} v_{x_1x_2\ldots x_D}$$

where $\underset{d}{E}$ is the translation operator [24, p. 77] and is defined by:

$$\underset{d}{E}\, v_{x_1x_2\ldots x_d\ldots x_D} = v_{x_1x_2\ldots x_d+1\ldots x_D}$$

(**1** is the identity operator). Applying the binomial theorem and multiplying series, the mixed difference operator becomes:

$$\sum_{y_1=0}^{z_1}\sum_{y_2=0}^{z_2}\cdots\sum_{y_D=0}^{z_D}(-1)^{(\Sigma z_j-\Sigma y_j)}\binom{z_1}{y_1}\binom{z_2}{y_2}\cdots\binom{z_D}{y_D}u_{x_1+y_1,x_2+y_2,\ldots,x_D+y_D}$$

$$=\sum_{y_1=0}^{n_1}\sum_{y_2=0}^{n_2}\cdots\sum_{y_D=0}^{n_D}(-1)^{(\Sigma z_j-\Sigma y_j)}\binom{z_1}{y_1}\binom{z_2}{y_2}\cdots\binom{z_D}{y_D}v_i$$

where

$$i = \phi(x_1 + y_1, x_2 + y_2, \ldots, x_D + y_D). \qquad \text{(D1)}$$

The upper limits on the sums are increased from z_d to n_d, for $d = 1, 2, \ldots, D$ (respectively), so that all terms are generated. This can be done because the additional terms introduced are zero due to the definition of the binomial coefficient. The mixed difference is positive when $x_d=1, 2, \ldots, n_D-z_D$ for *all* $d=1, 2, \ldots$, and is zero otherwise. Also, the mixed difference is zero if $y_d<0$ or $y_d>z_d$ for any $d=1, 2, \ldots, D$.

These mixed differences can be calculated by multiplying the vector **v** on the left by the matrix:

$$M^0 = \{m^0_{st}\}_{\substack{s=1,2,\ldots,n\\ t=1,2,\ldots,n}} \quad ; n = n_1n_2\ldots n_D.$$

The quantity m^0_{st} depends on the values of both $\underline{x}=(x_1, x_2, \ldots, x_D)$ and $\underline{y}=(y_1, y_2, \ldots, y_D)$ in Equation (4). The value of **x** is determined by:

$$\mathbf{x} = (x_1, x_2, \ldots, x_D) = \mathbf{\Phi}^{-1}(s),$$

where s is the 1-row coordinate of m^0_{st}.The value of **y** is determined by:

$$\mathbf{y} = (y_1, y_2, \ldots, y_D) = \mathbf{\Phi}^{-1}(t - s + 1) - 1,$$

where t is the coordinate of m_{st}^{0}, which is defined as:

$$m_{st}^{0} = \begin{cases} (-1)^{(\Sigma z_p - \Sigma y_p)} \binom{z_1}{y_1} \binom{z_2}{y_2} \dots \binom{z_D}{y_D}; & \text{if } x_d \leq n_d - z_d \text{ for all } d = 1, 2, \dots, D \\ 0\,; & \text{if } x_d > n_d - z_d \text{ for any } d. \end{cases}$$

Note that m_{st}^{0} if $y_d > z_d$ or $y_d < 0$ (for any d) since

$$\binom{z_d}{y_d} = 0$$

by definition. As a result $m_{st}^{0} = 0$ for $t < s$. The matrix $\mathbf{M}^{1}$ is defined as $\mathbf{M}^{1} = \{m_{st}^{1}\}$, where

$$m_{st}^{1} = \begin{cases} (-1)^{(\Sigma z_j - \Sigma y_j - D)} \binom{z_1 - 1}{y_1} \binom{z_2 - 1}{y_2} \dots \binom{z_D - 1}{y_D}; & \text{if } x_d \leq n_d - z_d \\ & \text{for all } d = 1, 2, \dots, D \\ 0\,; & \text{if } x_d > n_d - z_d \\ & \text{for any } d. \end{cases}$$

and the all-zero rows are the same as those of M^0. Also, $m_{st}^{1} = 0$ if $y_d > z_d - 1$ or $y_d > 0$ (for any d), since

$$\binom{z_d - 1}{y_d} = 0$$

by definition.

Then S_2 becomes

$$\begin{aligned} S_2 &= k \sum_{x_1=1}^{n_1 - z_1} \sum_{x_2=1}^{n_2 - z_2} \dots \sum_{x_D=1}^{n_D - z_D} \left[\underset{1}{\Delta}^{z_1} \underset{2}{\Delta}^{z_2} \dots \underset{D}{\Delta}^{z_D} - r \underset{1}{\Delta}^{z_1 - 1} \underset{2}{\Delta}^{z_2 - 1} \dots \underset{D}{\Delta}^{z_D - 1} v_i \right] \\ &= [(M^0 - rM^1)v]^T (M^0 - rM^1)v \\ &= v^T M^T M v \end{aligned}$$

where

$$M = M^0 - rM^1.$$

APPENDIX E

ADDITIONAL DETAILS ON VECTORIZATION

A multidimensional object is "vectorized" as follows [30]:

$$\begin{aligned}
v_1 &= u_{1,1,\ldots,1,1} \\
v_2 &= u_{1,1,\ldots,1,2} \\
&\vdots \\
{}_{n_D} &= u_{1,1,\ldots,1,D} \\
{}_{n_{D+1}} &= u_{1,1,\ldots,2,1} \\
&\vdots \\
v_{n_{D-1}n_D} &= u_{1,1,\ldots,n_{D-1},n_D} \\
&\vdots \\
v_P &= u_{n_1,n_2,\ldots,n_D}.
\end{aligned}$$

Note that the last index is increased first.

Then **m** can be written as

$$\mathbf{m} = (m_1, m_2, \ldots, m_{D-1}, m_D)$$

where

$$m_i = \prod_{j=i+1}^{D} n_j; \quad m_D = \prod_{D+1}^{D} n_j \equiv 1.$$

There is a unique inverse function denoted by: $\phi^{-1}(i)=(x_1, x_2, \ldots, x_D)$. The function $\phi^{-1}(i)$ is calculated by dividing $i-1$ by n_D. The remainder is x_D-1. The resulting quotient is divided by n_{D-1}. This remainder is $x_{D-1}-1$, and so on.

DISCUSSION OF PRECEDING PAPER

ELIAS S.W. SHIU:

Professor Lowrie is to be thanked for this comprehensive paper on multidimensional graduation. I would like to raise the issue whether it is still appropriate to have Henderson's name attached to the method. Perhaps some might even argue that Henderson's contribution had actually mystified the Whittaker graduation and retarded its development. Here is the first paragraph of Chapter 5 of Miller's monograph [3].

> 5.0 Professor E. T. Whittaker of the University of Edinburgh first enunciated the principles of the difference-equation method in a paper published in 1919. Subsequently, Robert Henderson developed a practical process for employing the method to make a numerical graduation. For these reasons, difference-equation formulas are also referred to as Whittaker-Henderson formulas.

Three pages later, Miller [3] explains Henderson's contribution.

> 5.3 *The practical method of solving the difference equation.*
>
> Formula A is used to graduate a great many other series than rates of mortality. For this reason, (5.21) is usually written in terms of u_x's rather than q_x's. The plain u_x's refer to the graduated series, the u_x'''s to the ungraduated series. In terms of these u_x's equation (5.21) becomes
>
> $$u_x'' = u_x + h\delta^4 u_x. \qquad (5.31)$$
>
> The Henderson process for solving (5.31) is based on the fact that the equation may be factored in terms of its finite-difference operators and, by the introduction of the intermediate series, u_x', be replaced by two simpler difference equations:
>
> $$u_x'' = \tfrac{1}{2}(a + 1)(a + 2)u_x' - a(a + 2)u_{x-1}' + \tfrac{1}{2}a(a + 1)u_{x-2}', \qquad (5.32a)$$
>
> $$u_x' = \tfrac{1}{2}(a + 1)(a + 2)u_x - a(a + 2)u_{x+1} + \tfrac{1}{2}a(a + 1)u_{x+2}; \qquad (5.32b)$$
>
> provided h and a are connected by the relationship
>
> $$h = \tfrac{1}{4}a(a + 1)^2(a + 2). \qquad (5.33)$$
>
> The actual factorization of (5.31) is demonstrated in (10.7).

> That the two equations (5.32a) and (5.32b) are equivalent to (5.31) may be verified by substituting the u_x''s obtained from (5.32b) in (5.32a).
>
> These two second-order difference equations are used in the form
>
> $$u_x' = \frac{2a}{a+1}u_{x-1}' - \frac{a}{a+2}u_{x-2}' + \frac{2}{(a+1)(a+2)}u_x'', \quad (5.34a)$$
>
> $$u_x = \frac{2a}{a+1}u_{x+1} - \frac{a}{a+2}u_{x+2} + \frac{2}{(a+1)(a+2)}u_x'. \quad (5.34b)$$
>
> If two u_x''s at the zero end are supposed to be known, all the subsequent u_x''s may be found, in order, by (5.34a). If two u_x''s at the other end can be found, the rest of the graduated series may be derived, in reverse order, by (5.34b). The only further problem that must be solved is the determination of these four "starting values."
>
> It may be shown (see (10.8)) that the two u_x's at the ω-end can always be found accurately as soon as the intermediate u_x' series has been calculated. They are
>
> $$\begin{aligned} u_{\omega-1} &= u_{\omega-1}' + a\Delta u_{\omega-1}', \\ u_\omega &= u_\omega' + a\Delta u_{\omega-1}'. \end{aligned} \quad (5.35)$$
>
> At the zero end of the series, the two needed values of u_x' cannot be determined accurately at the outset except by involved methods. If such methods are not to be resorted to, the first two graduated values must be estimated from the general run of the ungraduated values at the zero end. The graduation is completed using these estimated values and then corrected, if necessary.

I think one reason for the high esteem of the Whittaker graduation in North America has been the complexity of Henderson's solution. The above is merely a description of the solution for the Type A graduation, without the technical details given in the Appendix [3, Sections 10.7, 10.8 and 10.9]. Miller [3, p. 40, p. 56] wrote: "Formula B is rarely used except for making graduations of considerable importance. . . . The Type B formulas, however, require a greater degree of technical knowledge and substantially more difficult arithmetical work, which tend to restrict their use to the construction of important tables."

Although Miller [3, p. 40] stated that "[t]he method of solution [for the Whittaker graduation] involves the solution of ω+1 [linear] equations in $\omega+1$ unknowns," he did not seem to have searched for a more effective solution from the numerical linear algebra literature. In 1955 Cragoe [1] did point out that the Choleski factorization algorithm could be used instead of the Henderson algorithm, but this important observation seemed not to have been understood by most actuaries until the appearance of Greville's Study Note in 1973.

I would like to repeat McKay's [2] statement that "any discussion of advances in Whittaker-Henderson graduation techniques should give credit to Dr. T.N.E. Greville, whose lucid Part 5 Study Note was a great help to a decade of actuarial students." I would further add that Greville's Study Note was also a great help to actuarial researchers, as evidenced by comparing the number of papers on graduation theory that appeared before and after the publication of the Study Note.

Let me end this discussion with a philosophical question. The Whittaker graduation is a global method of graduation. By "global" I mean that the method uses all observed values simultaneously for obtaining the graduated values. Changing one observed value will change all graduated values, whether they are of ages close by or far away. Is this reasonable?

REFERENCES

1. CRAGOE, A.C. Discussion of CAMP, K. "New Possibilities of Graduation," *TSA* VII (1955): 413–15.
2. MCKAY, S.F. Discussion of KNORR, F. "Multidimensional Whittaker-Henderson Graduation," *TSA* XXXVI (1984): 241.
3. MILLER, M.D. *Elements of Graduation*. New York, N.Y.: Actuarial Society of America and American Institute of Actuaries, 1946.

(AUTHOR'S REVIEW OF DISCUSSION)

WALTER B. LOWRIE:

I thank Dr. Shiu for his comments. I agree that Henderson's name should be omitted from this method of graduation. I used it from habit. The comment about the effects of a change in one datum on the entire graduation is appropriate. This should be a matter of future research.