Lawrence Berkeley National Laboratory

Recent Work

Title

SMOOTHING DATA WITH TOLERANCES BY USE OF LINEAR PROGRAM-MING

Permalink

<https://escholarship.org/uc/item/62b8h6gq>

Author

Young, Jonathan D.

Publication Date

1970-11-01

SMOOTHING DATA WITH TOLERANCES BY
USE OF LINEAR PROGRAMMING

RECEIVED
LAWRENCE
RADIATION LABORATORY

Jonathan D. Young

FEB 9 1971

November 1970

LIBRARY AND
DOCUMENTS SECTION

AEC Contract No. W-7405-eng-48

TWO-WEEK LOAN COPY

*This is a Library Circulating Copy
which may be borrowed for two weeks.
For a personal retention copy, call
Tech. Info. Division, Ext. 5545*

32 LAWRENCE RADIATION LABORATORY
UNIVERSITY of CALIFORNIA BERKELEY

DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor the Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or the Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or the Regents of the University of California.

SMOOTHING DATA WITH TOLERANCES BY
USE OF LINEAR PROGRAMMING

Jonathan D. Young

Lawrence Radiation Laboratory
University of California
Berkeley, California 94720

November 1970

ABSTRACT

This report describes a method for smoothing observed data when the values of the dependent variable are assumed to lie within specified tolerance of the observed value. Linear programming is employed to obtain optimal smoothing of the (local) degree desired within the restraints imposed by the tolerances. First- and third-degree smoothing are discussed, but the method may be extended to higher order.

I. INTRODUCTION

A function f and t has been measured for n distinct and increasing values, t_i , providing f_i^* . The t_i need not be uniformly spaced. For each i , it is assumed that the true value f_i for f lies within some specified tolerance of the measured value f_i^* . That is, the measured value f_i^* may be too large by a quantity d_i^- or too small by a quantity d_i^+ , with d_i^- and d_i^+ greater than or equal zero:

$$\begin{aligned} f_i &\geq f_i^* - d_i^- & \text{for } i = 1 \cdots n; \\ f_i &\leq f_i^* + d_i^+ & \text{for } i = 1 \cdots n. \end{aligned}$$

The tolerances, d_i^- and d_i^+ , need not be uniform for the various i and may be zero for some i . This freedom is compatible with measurements known to be more precise at some points than at others. It is also consistent for measurements known to be maximum or minimum values of f at some of the t_i .

Milne¹ proposes smoothing the data, (f_i^*, t_i) for $i = 1 \cdots n$ to obtain f_i by computing a polynomial of degree $m < n$ by the method of least squares. The "smoothed" values, f_i , are then the values of the polynomial at the points t_i . For data for which tolerances are uniform and equal in each direction.

$$\begin{aligned} d_i^- &= d_{i+1}^- & \text{for } i = 1 \cdots n-1; \\ d_i^- &= d_i^+ & \text{for } i = 1 \cdots n; \end{aligned}$$

then for some degree, m , the values of f_i obtained will lie within tolerance. This is certainly true for $m = n - 1$, in which case the polynomial is an exact fit for (f_i^*, t_i) . In practice, we might seek the

least m for which all tolerances are met. For tolerances that are nonuniform (but still equal in each direction at each t_i), the errors at each t_i can be weighted in effecting the least-square fit, thereby adjusting the fit to suit the tolerance. The difficulty of nonequal tolerances can be handled by first moving f_i^* to the midpoint of the allowable variation, obtaining

$$f_i^{**} = f_i^* + \frac{d_i^+}{2} - \frac{d_i^-}{2}.$$

The least-square fit will then assume that f_i^{**} is the most preferred value for f_i , which may or may not be true. If the required degree m to achieve fit within tolerance is relatively large, the values f_i obtained may be based on more inflection in the curve for $f(t)$ than the observer with some a priori knowledge will accept. Interpolation using the least-square polynomial often provides evidence of such inflection. Thus the very globality whereby the change $f_i^* \rightarrow f_i$ is affected by all (t_i, f_i^*) may produce unacceptable results for some data.

In contrast to the least-square polynomial, smoothing is the filter method described by Hamming.² He sets

$$f_i = \sum_{j=1}^n a_j f_j^*, \quad \text{with} \quad \sum a_j = 1,$$

and requires $a_j = 0$, except for a few indices near and including i . For example, for uniformly spaced t , we might set

$$f_1 = (5f_1^* + 2f_2^* - f_3^*)/6$$

$$f_i = (f_{i-1}^* + f_i^* + f_{i+1}^*)/3, \quad \text{for } i = 2 \dots n - 1.$$

$$f_n = (-f_{n-2}^* + 2f_{n-1}^* + 5f_n^*)/6,$$

This choice of coefficients, a_j , is derived from the least-square first-degree approximation over three points described by Hildebrand,³ who also gives formulas for third- and fifth-degree approximations over five and seven points respectively. Obviously the filter method is "local", since only f_i^* and a few nearby values affect the resulting f_i . Some global effect may be introduced by repeating the smoothing process on the new values.

For nonuniform steps in t , the coefficients a_j above may be appropriately modified. In general, odd-degree fits are used due to the resulting symmetry in the formulation for interior points. In general, the higher the degree the less smoothing will take place; in fact for degree $n - 1$, there will be no smoothing, since the polynomial will be an exact fit of the data. Without tolerances, the choices of degree (perforce less than $n - 1$) and the number of iterations of the smoothing are subjective, possibly based on a priori knowledge of the behavior of f . Iteration of the smoothing process with approximation of degree m on $m + 2$ points results in the limit in a polynomial of degree m for the whole set of data, which may or may not be desirable.

When tolerances are specified, one might choose the least degree-point approximation, so that after one smoothing, the values f_i remain within tolerance. One might iterate the smoothing as long as

the resulting values remain within tolerances. As an alternative, a higher degree-point approximation might be employed, again with (presumably more) iterations on the smoothing. The end sought is a proper balance between local and global effect, which is a subjective (possibly well-founded) decision. A practical consideration is the amount of computation involved, which is less for a lower degree.

In the problem of constructing an interpolating function for f based on (t_i, f_i) for $i = 1 \cdots n$, valid for the whole interval $(t_1 \cdots t_n)$, a good balance between local and global effects may often be attained by employing the cubic spline fit.⁴ This function $s(t)$, defined on $(t_1 \cdots t_n)$, is made up of cubic (in t) arcs defined on $(t_i \cdots t_{i+1})$, where $i = 1 \cdots n-1$, with the requirements of $s_i(t_i) = f_i$ and the matching of first and second derivatives of the left and right cubic segments at each interior point t_i , where $i = 2 \cdots n-1$. Since successive cubics are, in general, distinct, the function s will not possess a third derivative at the interior points, t_i with $i = 2 \cdots n-1$. At these points we have left and right third derivatives which are not, in general, equal. We may use magnitudes of the differences between left and right third derivatives as a measure of the discrepancy of the fit.

For $n \geq 5$, when it is intended to use the cubic spline as the interpolating function on the smoothed values (t_i, f_i) , it seems obvious that third-degree five-point smoothing and iteration thereof will tend to reduce the discrepancy of the cubic spline fit. In the limit of iteration the smooth f will be a cubic and there will be no discrepancy. This result may or may not be desirable, depending on a priori knowledge.

When tolerances are imposed, the values f_i obtained by the third-degree five-point approximation may all lie within these tolerances. Then we can iterate the smoothing as long as resulting values, f_i , are still acceptable. On the other hand, some of the f_i may fall outside tolerance and remain so, and this approximation cannot be used. Rather than going to a higher degree, it may be desirable to devise a smoothing process as near to the third-degree approximation as possible while still obtaining smoothed values within tolerances.

II. FIRST-DEGREE SMOOTHING

We now consider a problem, similar to but simpler than that posed in the preceding section. For a set of data $(t_i, f_i^*, d_i^-, d_i^+)$, with $i = 1 \dots n$, for $n \geq 3$ we wish to find a smoothed set, (t_i, f_i) , so that for $i = 2 \dots n-1$ the f_{i-1} , f_i , and f_{i+1} lie "nearly" on a straight line with

$$\left. \begin{array}{l} f_i \geq f_i^* - d_i^- \\ f_i \leq f_i^* + d_i^+ \end{array} \right\} \quad \text{with } i = 1 \dots n.$$

We can employ the first-degree three-point approximation described earlier. If the first smoothing yields any f_i outside tolerance and $n = 3$, then no solution within tolerance is possible by this method, since iteration will not change the result. However, for $n \geq 4$, it is possible that some successive iteration may bring the smoothed values within the tolerances. When or whether this will occur is difficult to predict. Our experience with rather varied data indicates that if it does not occur after a few iterations it will not occur, and on solution within tolerance is possible.

On the other hand, if the first smoothing yields all f_i within tolerance, and the points (t_i, f_i) lie on a straight line, we have already achieved a desired smooth set as nearly (in fact on) a straight line as possible, and we have all second divided differences equal to zero or equivalently (without the second division)

$$\epsilon_i \equiv \frac{1}{h_i} f_{i+1} - \left(\frac{1}{h_i} + \frac{1}{h_{i-1}} \right) f_i + \frac{1}{h_{i-1}} f_{i-1} = 0 \quad \text{for } i = 2 \cdots n-1$$

with $h_i = t_{i+1} - t_i$.

On the contrary, if the smoothed f_i are all within tolerance but the (t_i, f_i) are not on a straight line, then certainly for some i from $i = 2 \cdots n-1$ we have $\epsilon_i \neq 0$, and we can use the magnitude of the ϵ_i to define "nearness" to a straight line.

The following norms of overall nearness may be considered:

$$\begin{aligned} \mu_f &= \max_{2 \cdots n-1} |\epsilon_i| \\ \nu_f &= \sum_{i=2}^{n-1} |\epsilon_i|, \\ \sigma_f &= \left[\sum_{i=2}^{n-1} \epsilon_i^2 \right]^{1/2}. \end{aligned}$$

Each of these norms is zero if the (t_i, f_i) lie on a straight line. Further, since iteration of the smoothing process tends to that result, we may assume that each of the norms tends to zero with increasing iteration. Thus, as long as smoothed values are found in tolerance, we can iterate and expect to find results nearer to a straight line. We

can choose one of the norms above and iterate until there is no significant change in its value and assume that the final values f_i obtained are in fact optimum (that is, as nearly on a straight line as is possible with this method).

The method described above may fail on two counts:

- (a) it may never find a solution within tolerance,
- (b) the value of the norms attained by it may not be the smallest possible.

These failures are due to the fact that the criterion of the process is minimization (in the least-square sense) of smoothing rather than minimization (in any sense) of undivided second differences. (See Examples 1 and 2, Section IV.)

We therefore propose a method which has the latter minimization as its objective. However, since the third norm, σ_f , does not lead to a linear model, we shall leave it for future development. Either of the other measures leads to a linear model. We elect to use v_f , since it has more global character.

We have the following linear restraints:

$$\left. \begin{aligned} f_i &\geq f_i^* - d_i^- \\ -f_i &\geq -f_i^* - d_i^+ \end{aligned} \right\} \text{ for } i = 1 \cdots n,$$

$$\left. \begin{aligned} \frac{1}{h_i} f_{i+1} - \left(\frac{1}{h_i} + \frac{1}{h_{i-1}} \right) f_i + \frac{1}{h_{i-1}} f_{i-1} + \alpha_i &\geq 0 \\ -\frac{1}{h_i} f_{i+1} + \left(\frac{1}{h_i} + \frac{1}{h_{i-1}} \right) f_i - \frac{1}{h_{i-1}} f_{i-1} + \alpha_i &\geq 0 \end{aligned} \right\} \text{ for } i = 2 \cdots n-1,$$

$$\alpha - \sum_{i=2}^{n-1} \alpha_i \geq 0$$

where $\alpha_i \geq |\epsilon_i| \geq 0$, and $\alpha \geq \sum_{i=2}^{n-1} \alpha_i \geq \nu_f$.

We let $w = \alpha$.

Subject to the restraints above, we can determine values of f_i for $i = 1 \dots n$ so as to minimize w and consequently ν_f .

We define

$$\vec{f} \equiv (f_1, \dots, f_n, \alpha, \alpha_2, \dots, \alpha_{n-1}).$$

Then the above leads to a linear model of the form

$$\text{Minimize } w = \vec{b} \cdot \vec{f},$$

$$\text{subject to } B\vec{f} \geq \vec{c},$$

where B is a sparse matrix with $4n-3$ rows and $2n-1$ columns, $\vec{c}$ has $4n-3$ components, and $\vec{b}$ has $2n-1$ elements, all of which are zero except for the $n+1$ component (corresponding to α), which is one. This model always has a solution, since in any case $w \geq 0$; in the absence of any other solution we could set $f_i = f_i^*$, and if this gives $w = 0$, then the observed points (x_i, f_i^*) lie on a straight line and no smoothing is needed. Otherwise, $w(f_i^*) \equiv w^*$ has a finite value and there is a minimum w ,

$$0 \leq w \leq w^*.$$

The solution in terms of f_i is not necessarily unique--that is, the minimum value of w may be attained for more than one set of values of the f_i .

The linear model described above is not a linear program, since the f_i are not restricted to nonnegative values. In theory, it is advantageous to remove this deficiency by introducing new translated

variables (perforce nonnegative) to replace the f_i and then translate back after an optimal solution is obtained. We employ an alternative which we believe is better in practice. (See second paragraph below.)

The dual of the above model has the form

$$\begin{aligned} \text{Maximize} \quad z &= \vec{c} \vec{y}, \\ \text{subject to} \quad A \vec{y} &= \vec{b} \\ \text{with} \quad y_j &\geq 0, \quad \text{for } j = 1 \cdots 2n-1, \end{aligned}$$

where A is the transpose of B and y is the (formal) dual variable vector.⁵ This transposed model is a linear program and can be solved by the Simplex algorithm.⁶ By the duality theorem

$$\text{Min } w = \text{Max } z,$$

hence the program does have a solution. In addition to providing us with a solution to the linear program, the Simplex algorithm also provides us with optimal values for its dual variables. These latter are precisely the values of f_i sought, together with values of α and the α_i therefor.

In justification of the duality approach, we recall that the amount of computation in the Simplex algorithm is dependent more on the number of restraints than on the number of variables. The dual model has fewer $(2n-1)$ restraints than does the original $(4n-3)$. Further, in the dual formulation, no slack variables need be introduced, since the restraints are equalities. During the computation, primal variables are rearranged, whereas dual variables retain their proper order throughout. Some computer codes do not restore the primal variables to proper order prior to output, leaving this for the user to do by hand.

By using the dualization, we eliminated the translation. These considerations have led to our choice of dualizing the model.

From the above, we now have a solution as near (in the sense of minimum v_f) as possible to a straight line. Recalling that this solution is not necessarily unique, we may now seek a solution which retains this property (the same v_f) but which also minimizes the "magnitude" of the smoothing. We let

$$c_i = f_i - f_i^* \quad \text{for } i = 1, \dots, n$$

and consider the following norms of the amount of smoothing:

$$\mu_c = \max |c_i|,$$

$$v_c = \sum_{i=1}^n |c_i|,$$

$$\sigma_c = \left[\sum_{i=1}^n c_i^2 \right]^{1/2}.$$

For reasons previously stated, we elect to minimize v_c . We retain all of our original restraint matrix A and add the restraint

$$-\alpha \geq -v_f,$$

where v_f is the minimal value attained above (note that α is restricted to the value v_f) and

$$\left. \begin{aligned} \gamma_i + f_i &\geq f_i^* \\ \gamma_i - f_i &\geq -f_i^* \end{aligned} \right\} \quad i = 1, n,$$

$$\gamma - \sum_{i=1}^n \gamma_i \geq 0,$$

where

$$\gamma_i \geq |c_i| \geq 0 \quad \text{and} \quad \gamma \geq \sum_{i=1}^n \gamma_i.$$

As our objective we set $w = \gamma$ and seek to minimize w , hence γ , hence v_c subject to the restraints.

Dualization and solution of the resulting linear program will provide us now with values for f_i for $i = 1 \dots n$ such that (t_i, f_i) lie as nearly (in the sense of minimum v_f) as possible on a straight line, and such that the smoothing is as little (in the sense of v_c) as possible therefor.

A numerical example of first-degree smoothing is given in Section IV.

III. THIRD-DEGREE SMOOTHING

For a set of data $(t_i, f_i^*, d_i^-, d_i^+)$ with $i = 1 \dots n$ for $n \geq 5$, we wish to find a smoothed set (t_i, f_i) so that for $i = 3 \dots n-2$, the f_{i-2} , f_{i-1} , f_i , f_{i+1} , and f_{i+2} lie "nearly" on a cubic arc with

$$\begin{aligned} f_i &\geq f_i^* - d_i^-, \\ f_i &\geq f_i^* + d_i^+. \end{aligned}$$

Third-degree five-point approximation can be employed. For $n = 5$, if the first smoothing is not within tolerance, no such solution is possible by this method. For $n \geq 6$, our experience has been that if a smoothed set within tolerance does not occur within a few iterations, it is not likely to occur ever.

On the other hand, if the first smoothing yields all f_i within tolerance and all the points (t_i, f_i) lie on a cubic arc from t_1 to t_n , we have already achieved our object, and we have all fourth divided differences equal to zero or equivalently (without the last division)

$$\begin{aligned} \epsilon_i = & \left[\frac{1}{q_{i-1} p_i h_{i+1}} \right] f_{i+2} - \left[\frac{1}{q_{i-1}} \left(\frac{1}{p_i h_{i+1}} + \frac{1}{p_i h_i} + \frac{1}{p_{i-1} h_i} \right) + \frac{1}{q_{i-2} p_{i-1} h_i} \right] f_{i+1} \\ & + \left[\frac{1}{q_{i-1}} \left(\frac{1}{p_i h_i} + \frac{1}{p_{i-1} h_i} + \frac{1}{p_{i-1} h_{i+1}} \right) + \frac{1}{q_{i-2}} \left(\frac{1}{p_{i-1} h_i} + \frac{1}{p_{i-1} h_{i-1}} + \frac{1}{p_{i-2} h_{i-1}} \right) \right] f_i \\ & - \left[\frac{1}{q_{i-1} p_{i-1} h_{i-1}} + \frac{1}{q_{i-2}} \left(\frac{1}{p_{i-1} h_{i-1}} + \frac{1}{p_{i-2} h_{i-1}} + \frac{1}{p_{i-2} h_{i-2}} \right) \right] f_{i-1} \\ & + \left[\frac{1}{q_{i-2} p_{i-2} h_{i-2}} \right] f_{i-2} = 0 \quad \text{for } i = 3 \cdots n-2, \end{aligned}$$

where

$$\begin{aligned} p_i &= h_i + h_{i+1} & \text{for } i = 1 \cdots n-2, \\ q_i &= h_i + h_{i+1} + h_{i+2} & \text{for } i = 1 \cdots n-3. \end{aligned}$$

On the contrary, if the smoothed values are within tolerance but not on a cubic, then some ϵ_i from $i = 3 \cdots n-2$ is not zero, and we can define norms of overall nearness by

$$\mu_f = \max_{i = 3 \cdots n-2} |\epsilon_i|,$$

$$\gamma_f = \sum_{i=3}^{n-2} |\epsilon_i|,$$

$$\sigma_f = \left[\sum_{i=3}^{n-2} \epsilon_i^2 \right]^{1/2}.$$

Iteration may be employed as for the first degree.

The method may fail for the reasons given for the first-degree smoothing, hence we are led again to employ linear programming.

We retain the tolerance restraints

$$\left. \begin{aligned} f_i &\geq f_i^* - d_i^- \\ -f_i &\geq -f_i^* - d_i^+ \end{aligned} \right\} \quad \text{for } i = 1 \dots n$$

and add

$$\left. \begin{aligned} \alpha_i - \epsilon_i &\geq 0 \\ \alpha_i + \epsilon_i &\geq 0 \end{aligned} \right\} \quad \text{for } i = 3 \dots n-2,$$

where ϵ_i is written in terms of f_{i+2} , f_{i+1} , f_i , f_{i-1} , and f_{i-2} as given earlier:

$$\alpha - \sum_3^{n-2} \alpha_i \geq 0 \quad \text{for } i = 3 \dots n-2,$$

where

$$\alpha_i \geq |\epsilon_i| \geq 0 \quad \text{and} \quad \alpha \geq \sum_3^{n-2} \alpha_i.$$

Thereafter the procedure is analogous to the first-degree case with suitable adjustment to provide for differing dimensions; for example,

$$\vec{f} \equiv (f_1 \dots f_n, \alpha, \alpha_3 \dots \alpha_{n-2})$$

and has $2n-3$ components and the restraint matrix B has $4n-7$ rows and $2n-3$ columns.

The subsequent "minimization of smoothing" is analogous to the first-degree process.

We are also interested in how the smoothing processes affect the cubic spline fit. We have claimed that the discontinuity of third derivatives at internal points will be reduced. The following norms can be considered:

$$\mu_e = \max_{i=2, n-1} |e_i|,$$

$$\nu_e = \sum_{i=2}^{n-1} |e_i|,$$

$$\sigma_e = \left[\sum_{i=2}^{n-1} e_i^2 \right]^{1/2},$$

where e_i is the difference between the left and right third derivatives of the cubic spline fit at the interior points, t_i for $i = 2 \cdots n-1$.

In all examples tried, an optimal cubic spline of rough data (not on a cubic) and smooth data revealed a significant reduction of all three norms. (We define an optimal cubic spline as one in which terminal second derivatives are chosen to minimize (in the least-square sense) the third-derivative discontinuities. (See Reference 7.) Some of these examples are given in the next section.

IV. ILLUSTRATIVE NUMERICAL EXAMPLES

We present here a few simple examples which illustrate the smoothing methods discussed and confirm that the linear program method can give useful results when other methods fail to meet tolerance requirements.

In the interest of simplicity, uniform unit argument steps and, in the main, uniform tolerances are used with a small number of data points. We have, however, used the method with nonuniform steps, nonuniform tolerances, and more points with equal effectiveness.

In each example we first compute the global least-square fit, next the filter least-square smoothing with reasonable iteration, and finally linear program smoothing, first minimizing v_f and then v_c .

The results are given below with column designations as follows:

- A values of f_i from global least-square fit,
- B values of f_i from one filter smoothing,
- C(k) values of f_i after k filter smoothings,
- D values of f_i for minimum v_f within tolerances,
- E values of f_i for minimum v_f and v_c .

Values of the various norms discussed are given below the values of the corresponding f_i .

Example 1. First-degree smoothing over five points. Uniform tolerances: $d_i^- = d_i^+ = 0.6$, all $i = 1, \dots, 5$.						
t	f^*	A	B	C(10)	D	E
1.	1.	1.2	0.83333	0.99265	1.6	1.6
2.	2.	2.2	2.33333	2.01470	2.6	2.5
3.	4.	3.2	3.33333	3.02479	3.6	3.4
4.	4.	4.2	4.33333	4.01470	4.6	4.3
5.	5.	5.2	4.83333	4.99265	5.6	5.2
μ_f	2.	0.0	0.5	0.02017	0.0	0.0
v_f	4.	0.0	1.0	0.04410	0.0	0.0
σ_f	2.	0.0	0.70711	0.02633	0.0	0.0
μ_c	0.	0.8	0.66667	0.97521	0.6	0.6
v_c	0.	1.6	1.66667	1.01932	2.8	2.2
σ_c	0.	0.89443	0.84937	0.97549	1.26491	1.04881

The least-square fit (A) does not fall with tolerances. The filter fit (B) and (C) is converging slowly, whereas the value of μ_c is steadily increasing. It cannot produce a solution within tolerance.

The linear program for minimization of $v_f(D)$ does produce a line solution. The subsequent minimization of $v_c(E)$ improves this solution and, although within tolerance, compares favorably with the least-square result, which is not.

The last solution is not necessarily unique. If the amount of smoothing at the first point seems excessive, the tolerance here may be reduced. In fact, for $d_1^- = d_1^+ = 0.4$ and other tolerances unchanged, we obtain f_i : 1.4, 2.4, 3.4, 4.4, and 5.4 with σ_c reduced to 1.00 and all other norms unchanged.

Example 2. Third-degree smoothing over six points. Uniform tolerance $d_1^- = d_1^+ = 0.45$.

t	f^*	A	B	C(10)	D	E
1.	1.	0.95238	0.91429	0.92593	0.55	1.00
2.	2.	2.23810	2.34286	2.29630	2.15	2.30
3.	4.	3.52381	3.48571	3.55556	3.55	3.55
4.	4.	4.47619	4.51429	4.44444	4.45	4.45
5.	5.	4.76190	4.65714	4.70370	4.55	4.70
6.	4.	4.04762	4.08571	4.07407	3.55	4.00
μ_f	6.0	0.0	0.94286	0.00000	0.0	0.0
v_f	12.0	0.0	1.88571	0.00000	0.0	0.0
σ_f	8.48528	0.0	1.33340	0.00000	0.0	0.0
μ_c	0.0	0.47619	0.51429	0.44444	0.45	0.45
v_c	0.0	1.52381	1.88571	1.62963	2.40	1.50
σ_c	0.0	0.75593	0.88248	0.76264	1.01735	0.76485
μ_e	10.8	0.0	1.69715	0.00000	0.0	0.0
v_e	28.8	0.0	4.52572	0.00000	0.0	0.0
σ_e	16.	0.0	2.53000	0.00000	0.0	0.0

The least-square result (A) is not within tolerance. After ten iterations the filter fit is essentially cubic and stable with respect to

the c norms. It is within tolerance, hence may be considered satisfactory. The final linear program (E) shows slight reduction in v_c but some increase in σ_c . For the specified tolerance the linear program was not worth the effort. However, for any significantly smaller tolerance, only the linear program would give acceptable results.

On any of the results C(10), D, or E the cubic spline fit has no discontinuity of the third derivative, whereas for the rough-data discontinuities are considerable, as indicated by μ_e , v_e , and σ_e .

Example 3. Third-degree smoothing over seven points.
Relative tolerances $d_i^- = d_i^+ = 0.01 f_i^*$.

t	f^*	A	B	C(10)	D	E
0.	0.	-.01833	-.00263	-.00645	0.0	0.0
1.	1.	1.04648	1.01051	1.02582	1.01	1.01
2.	2.828	2.81098	2.81223	2.78927	2.79972	2.79973
3.	5.196	5.16600	5.19223	5.14890	5.20887	5.20959
4.	8.	8.00238	7.99820	7.98208	8.07717	8.08
5.	11.180	11.21095	11.18120	11.19195	11.24431	11.25140
6.	14.697	14.68255	14.69670	14.69401	14.55003	14.56420
μ_f	0.184	0.	0.10337	0.02575	0.	0.
v_f	0.249	0.	0.16579	0.05051	0.	0.
σ_f	0.19035	0.	0.11860	0.03114	0.	0.
μ_c	0.	0.04648	0.01577	0.04710	0.14697	0.1328
v_c	0.	0.15962	0.03599	0.15096	0.33960	0.33606
σ_c	0.	0.06970	0.01963	0.07000	0.18099	0.17383
μ_e	0.25996	0.	0.14941	0.03361	0.	0.
v_e	0.37996	0.	0.24393	0.05462	0.	0.
σ_e	0.27142	0.	0.16040	0.03649	0.	0.

The least-square solution fails to meet the tolerances at the first two points; the filter method fails at the first three after ten

iterations and is a long way from convergence. On the other hand, the linear program has found a cubic solution within tolerance.

A subsequent linear program computation with the relative tolerance reduced to $d_i^- = d_i^+ = 0.005 |f_i^*|$ gave the result

t	E		
0	0.	μ_f	0.11028
1.	1.005	ν_f	0.11028
2.	2.81386	σ_f	0.11028
3.	5.19822	μ_c	0.07349
4.	8.04	ν_c	0.17597
5.	11.22112	σ_c	0.09445
6.	14.62351	μ_e	0.16443
		ν_e	0.27218
		σ_e	0.17691

Even with this smaller tolerance, the cubic spline discrepancy has been reduced. However, the reduction scarcely seems worth the effort. Here the experimenter may decide to increase the tolerance to improve the result, to abandon smoothing and use the original data, or to fit the data in some way other than by cubic spline.

V. CONCLUSION

The linear program technique described herein is not suggested as a cure-all for smoothing difficulties, nor is it meant to supplant established methods where these give satisfactory results. We have found cases in which the least-square and filter methods result in a change of data values that is not acceptable. The imposition of tolerances can prevent this. In some of our examples, the linear program

provided a solution of the degree desired when both the other methods failed. In all fairness, we should acknowledge that a least-square algorithm can readily be devised that takes tolerances into account. This algorithm finds the solution of the degree desired within tolerances, provided such a solution exists. However, the process is non-linear and requires some (perhaps extensive) iteration. It would seem to have no computational advantage over the linear program method. The question still remains of data which cannot be smoothed within tolerance to a polynomial of the degree desired. In this case, if the filter method converges (with reasonable rapidity) to an acceptable result, then there is certainly no necessity for linear programming. Our experience with the filter method indicated that success with it is extremely fortuitous. On the contrary, the linear program technique always gives an optimum (in the sense of minimum v_f) solution. If this solution is not acceptable, we have, at least, done the "best" we could.

We have limited our discussion to first- and third-degree smoothing. The method can obviously be extended to apply to other polynomials. Our discussion of first-degree smoothing was primarily intended as introductory, although there certainly may be cases when an experimenter wants a line fit for his data. The motivation for the third-degree smoothing was the extensive interest at present in cubic spline fitting. We had hoped to smooth the data in such a way that the cubic spline fit would be smoother (in the sense of reducing third-derivative discontinuities). In the many cases tried in which such discontinuities existed, they were reduced. We are not prepared to say

that this reduction is optimum.

We believe that from the information presented herein, a reader with sufficient interest could prepare his own restraint matrix for the second degree or for degrees higher than the third. The former would be rather easy; however, for degrees higher than fifth, the formulation seems formidable.

VI. ACKNOWLEDGMENT

This work was done in part under the auspices of the U. S. Atomic Energy Commission.

REFERENCES

1. W. E. Milne, Numerical Calculus (Princeton University Press, Princeton, N. J., 1949), pp. 275-276.
2. R. W. Hamming, Numerical Methods for Scientists and Engineers (McGraw-Hill Book Company, N. Y., 1962), pp. 314-315.
3. F. B. Hildebrand, Introduction to Numerical Analysis (McGraw-Hill Book Company, N. Y., 1956), pp. 295-296.
4. G. Birkhoff and C. R. deBoor, "Piecewise Polynomial Interpolation and Approximation," in Approximation of Functions, H. L. Garabedian, Ed. (Elsevier Publishing Company, Amsterdam, 1956), pp. 164-168.
5. S. I. Gass, Linear Programming (McGraw-Hill Book Company, N. Y., 1958).
6. G. B. Dantzig, Rand Report P-89 (The Rand Corporation, Santa Monica, Cal., 1956).
7. J. D. Young, An Optimal Cubic Spline, in The Logistics Review, 6 (29), 33 (Winter 1970).

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

- A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or*
- B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.*

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

TECHNICAL INFORMATION DIVISION
LAWRENCE RADIATION LABORATORY
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA 94720