Lawrence Berkeley National Laboratory
Recent Work

Title
GENERAL CUBIC SPLINES

Permalink
https://escholarship.org/uc/item/0rd2r1cf

Author
Young, Jonathan D.

Publication Date
1975

GENERAL CUBIC SPLINES

Jonathan D. Young

January 1975

DISCLAIMER

GENERAL CUBIC SPLINES*

Jonathan D. Young
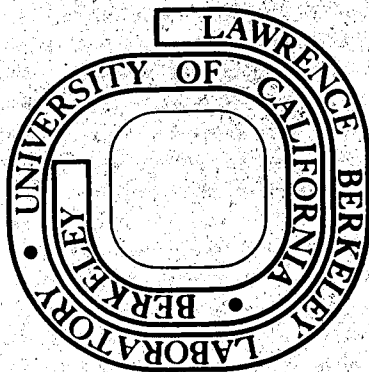
Lawrence Berkeley Laboratory
University of California
Berkeley, California 94720

January, 1975

ABSTRACT

We generalize the construction (computation of unknown function,

derivative and second derivative values) for a cubic spline with

specified knots subject to various sufficient conditions.

INTRODUCTION

In reference (1) we defined a cubic spline function on a closed interval, $[x_1, x_m]$, with knots, $x_1, x_2 \cdots x_m$. Briefly, the properties of the spline, $s(x)$, are:

   (1) On any subinterval, $[x_i, x_{i+1}]$, for i equals one to m - 1, s is a cubic in x.

   (2) The first and second derivatives, s' and s" are continuous over the whole interval, $[x_1, x_m]$.

In Property (1) the expression cubic includes degenerate cases. On the interval, $[x_i, x_{i+1}]$, the function is defined by

$$s = c_{i0} + c_{i1}\, x + c_{i2}\, x^2 + c_{i3}\, x^3 \quad \text{for x in } [x_i, x_{i+1}] \qquad (1)$$

Where any or all of the c's may be zero. Note also that the coefficients relate to the subinterval. This fact is indicated by the subscript, i.

   Now, for i equal one to m-2, we can set $j \equiv i + 1$ and write

$$s = c_{j0} + c_{ji}\, x + c_{j2}\, x^2 + c_{j3}\, x^3 \quad \text{for x in } [x_j, x_{j+1}] \qquad (2)$$

The function s need <u>not</u> be the same cubic on the consecutive intervals. We may have

$$c_{in} \neq c_{jn} \quad \text{for some or all n from zero to three.}$$

From the above, it would appear that we need 4m - 4 values for the c's in order to determine the spline, s. However, the c's are not independent since at any interior knot, x_j, j = 2, m - 1, function values, s_j, and by Property (2) first derivative values, s', and

and second derivative values, s", as expressed in terms of c_{in} and c_{jn} must match. Thus, we have three equations involving c's at each x_j for j = 2 to m - 1, for a total of 3m - 6 equations. From

$$(4m - 4) - (3m - 6) = m + 2$$

we see that only m + 2   c-values are independent. If an independent set of m+2 c-values are known, the rest can be computed. This fact agrees with the statement in reference (2) that the space of all cubic splines with m specified knots has dimension m + 2.

We are _not_ interested in the c-values and will not compute them. We are interested in the values of the cubic spline and its derivatives at each of the knots. We have _not_ required third derivative continuity. However, s is a cubic on any subinterval, hence, s'" is constant and continuous except for discontinuities ($c_{i3} \neq c_{j3}$) at interior knots, x_j. We can define a third derivative function (piecewise continuous, with discontinuities at interior knots, x_j) by assigning $s_j'"$, the value assumed from the right of x_j (the constant $6c_{j3}$). Fourth and all higher order derivatives are zero, as may be seen by differentiating either Equation (1) or Equation (2).

For all knots, x_i, for i = 1 to m, we shall call the 3m values, s_i, s_i' and $s_i"$ for i = 1 to m the primary values of the spline s with knots x_i, noting that s'" may be readily obtained by

$$s_i'" = (s_j" - s_i") / (x_j - x_i) \qquad \text{for } i = 1 \text{ to } m - 1$$

with j = i + 1
and $s_m'" = s_{m-1}'"$

From our previous remarks about independence of the c's and
dimension of the space of splines with m specified knots, we can
assume that if m + 2 independent primary values are specified, the
remaining

$$3m - (m + 2) = 2m - 2$$

values can be computed.

In reference (1) we showed that if the m function values, s_i,
for i = 1 to m and either s_1' and s_m' or s_1'' and s_m'' were specified,
then the remaining primary values could be computed. We were able
to do this in a rather special way: first computing all the unknown
s_i' by solving a "tridiagonal" linear system of m equations in m
unknowns and then computing all the unknown s_i'' one at a time using a
"two point" formula involving s_i, s_i', s_k and s_k' where x_k was adjacent
to x_i (either above or below). (NOTE we have m + 2 primary values
specified).

Now, suppose for a function, y(x), we know y_i for a set of m
distinct and increasing points, x_i for i = 1 to m, and we also know
y_1' and y_m' or y_1'' and y_m'' and the function is otherwise unknown. Then
the function y can be approximated by a cubic spline, s, whose values
are those known for y. We use the method of reference (1) to find all
the values, s_i' and s_i'', not known and then use these for values for
y_i' and y_i'' not known. Interpolation for approximate values for y, y'
and y" at any point x in $[x_1, x_m]$ can be performed by computing s, s',
and s" at that x.

However, the (two) kinds of specifications described above
are by no means the only sufficient ones to determine a unique
cubic spline.   Many other m + 2 primary values, provided one value
is specified at each knot and provided one value is a function value,
will determine a unique cubic spline. For example, a unique cubic
spline is determined if at _any_ knot x_j we specify s_j, s_j', and s_j'' and
at the other knots, x_i, $i \neq j$ we specify either s_i or s_i' or s_i''. As
another example specification of s_j and s_j' or s_j'' and s_k and s_k' or s_k'' at
knots, x_j, x_k and of s_i or s_i' or s_i'' at the other knots x_i, $i \neq j$,
$i \neq k$ is sufficient.

Unfortunately, the method of reference (1) cannot be readily
adapted to solving sufficient specifications of a general nature.
The usefulness of a more general method can be illustrated in
connection with function approximation. For example, if for y(x) we know
y_1, y_1' and y_1'' and y_i for i = 2 to m, there is a unique cubic spline
which approximates y but we cannot find it (compute its derivative
values) by the method of reference (1). As another example, suppose
we know y_i for i equal 1 to m, and further know that at interior knots,
x_j and x_k, y_j and y_k are extreme values (maximum or minimum), then we
can set $y_j' = 0$ and $y_k' = 0$ to sufficiently specify the unique cubic
spline s which approximates y. Further, if we know all the y_i and
know that between two consecutive knots x_j and x_k that y is a
straight line, then we can set $s_j'' = 0$ and $s_k'' = 0$ to specify the cubic
spline. Thus, the need for a general method becomes apparent.

GENERAL METHOD

We wish to devise a method which will compute $2m - 2$ unknown primary values (s_i, s_i', s_i'') from any sufficient specification of $m + 2$ primary values. We have $m - 1$ subintervals, $[x_i, x_j]$ where $j \equiv i + 1$ and i runs from one to $m - 1$. Now, on any of these subintervals, s is a cubic and, further, the value s_j, s_j', and s_j'' are those of that cubic. We set

$$d_i = x_j - x_i$$

and remembering that fourth and higher derivatives of a cubic are zero, we expand s, s' and s'' in Taylor's expansions from x_i to x_j

$$s_j = s_i + d_i s_i' + d_i^2 s_i''/2 + d_i^3 s_i'''/6 \tag{3}$$

$$s_j' = s_i + d_i s_i'' + d_i^2 s_i'''/2 \tag{4}$$

$$s_j'' = s_i'' + d_i s_i''' \tag{5}$$

Note that s_i''' is defined by "the right hand convention" mentioned earlier.

From Equation (5) we have

$$s_i''' = (s_j'' - s_i'')/d_i$$

and we replace s_i''' in Equations (3) and (4) by its expression in s_j'' and s_i'' obtaining (after some transposition)

$$s_i - s_j + d_i s_i' \qquad + d_i^2 s_i''/3 + d_i^2 s_j''/6 = 0 \tag{6}$$

$$s_i - s_j' \qquad + d_i s_i''/2 + d_i s_j''/2 = 0 \tag{7}$$

Now, Equations (6) and (7) hold for $i = 1$ to $m - 1$ and together provide us with $2m - 2$ equations involving $3m$ primary values of which $m + 2$ are

to be specified. Consequently, we have precisely 2m - 2 equations in
2m - 2 unknowns. When we evaluate the terms for those primary values
specified and transpose them, from Equations (6) and (7) applied for i + 1
to m - 1, we have   m - 2 equations of the form

$$(\text{sum of "unknown" terms}) = -(\text{sum of known terms}) \tag{8}$$

which we wish to solve for the unspecified primary values. The question
is - will this system be determinate (can we solve for the unknowns)?
Now, Equation (7) does not involve function values, hence, we have only
m - 1 equations [from (6)] which do. If all m function values are
unspecified, we have m unknowns appearing in only m - 1 equations.
Therefore, at least one function value must be specified (as stated
earlier) otherwise the system is indeterminate. Now suppose no primary
value is specified at x_1. These values appear in Equations (6) and (7)
only for i = 1 hence we have three unknowns appearing in only two equations
and the system is indeterminate. Similarly, at x_m, if no primary values
are specified, we have Equations (6) and (7) only for j = m, hence again,
only two equations for three unknowns. As for interior knots, there is
obviously no virtue in including a point x_j for which no value (function,
first derivative or second derivative) is known. Hence, we shall always
require that some primary value be known at every knot (at stated earlier).
Note that the requirements above are necessary conditions; they may
not be sufficient.

COMPUTATIONAL PROCEDURE

We are considering a cubic spline function, s, with knots, x_i for
i equal 1 to m with m greater than or equal to three. We must know the
values of the x_i and they must be distinct and in increasing order.
Of the primary values, s_i, s_i', s_i'' (3m in all) we must know m + 2 with
at least one function value known and at least one value known at each
knot. We set up an argument vector, x, whose components are the x_i of
dimension m, and an array, S, of dimension, 4 by m, whose rows are
(or are to be) values of s_i, s_i', s_i'' and s_i''' respectively for i equal
1 to m. Now, at the outset, we shall only know m + 2 of the primary
values, hence we set up an indicator array, K, of dimension three by m,
whose rows contain a known or unknown indication for s_i, s_i' and s_1''
respectively, (for convenience, the entry is to be zero if the value is
known and one if it is unknown). The values for the knots are placed
in the vector, x, the known primary values in the "value array", S, and
the corresponding indicators in indicator array, K.

We begin by counting (and numbering) the unknowns as indicated in
the indicator array. That is, we look at the first tow of the indicator
array. If the indicator is zero (known value) we do nothing, otherwise
the indicator which is one is replaced by a number for that unknown, i.e.
the first encountered is one, the second two, etc. We do the same for
the second row of the indicator - continuing the numbering, and then for
the third. If at any time the number for the unknown exceeds 2m - 2 we must
stop since there are too many unknowns. If we finish up with less than
2m - 2 unknowns, we also stop. Meanwhile, we can check that at least

one function value is known and that some value is known at every

knot, otherwise we stop.

We are now ready to construct the coefficient matrix, A, $(2m - 2)$ by

$(2m - 2)$ and right-hand-side $(2m - 2)$ for the linear system of

Equation (8). We now proceed for consecutive knots, (x_i, x_j) for i equal one to

m-1, and $j = i + 1$, computing $d_i = x_j - x_i$ and from d_i the coefficients for

Equations (6) and (7). Our numbering of the unknowns now becomes

important since that number is the column of the coefficient matrix

corresponding to that unknown. When the indicator is zero, the

value is known and the product of the coefficient and the known value

must be subtracted from the right-hand-side. Otherwise, the coefficient

is entered in the indicated column of the coefficient matrix, A. For each

i for $i = 1$ to $m - 1$ we obtain two rows for the coefficient matrix and

two entries for the right-hand-side. As we compute the d_i, we can

check that the x_i are distinct $(d_i \neq 0)$ and in proper order $(d_i > 0)$,

otherwise we must stop. Note that the coefficients of this paragraph

are those of Equations (6) and (7) not the c-values mentioned earlier.

We now have a linear system of the form

$$A \vec{p} = \vec{b}$$

A is $(2m - 2)$ by $(2m - 2)$, $\vec{p}$ is vector whose components are the

unknown primary values in the order described above and $\vec{b}$ is right-hand-side

above. Barring pathological circumstances (singular or ill-conditioned

matrix, A) this system can be solved for the unknown vector, $\vec{p}$. We now

again use our numbering system contained in the indicator array to

place the now known components of $\vec{p}$ in their appropriate places in

the value array.

We now compute s_i''' for i equal 1 to m - 1
$$j = i + 1$$

by $s_i''' = (s_j'' - s_i'')/d_i$

and set $s_m''' = s_{m-1}''$

and place these answers in the fourth row of the value array. Then

this array contains all the values for s.

ERROR ANALYSIS

Certainly there are pitfalls in approximating any function for which we know only enough values to specify a cubic spline. The specified values will, of course, be exact, but all other values computed for the spline may differ greatly from those of the function. Particularly, this is more likely when the two "extra" primary values (other than one per knot) are specified at one knot or at two knots close together. Hence, we need some method whereby the approximation error can be estimated.

The cubic spline approximation has zero fourth derivative values on every subinterval. In general, this will not be true for the function, y, being approximated. In order to get an estimate of errors, let us assume that s is in fact a quartic over the entire interval, $[x_1, x_m]$. Now, for y we can write for any $i = 1$ to $m - 1$, with $j \equiv i + 1$

$$y_j = y_i + d_i\, y_i' + d_i^2\, y_i''/2 + d_i^3\, y_i'''/6 + d_i^4\, y^{iv}/24 \tag{8}$$

$$y_j' = y_i' + d_i\, y_i'' + d_i^2\, y_i'''/2 + d_i^3\, y^{iv}/6 \tag{9}$$

$$y_j'' = y_i'' + d_i\, y_i''' + l_i^2\, y^{iv}/2 \tag{10}$$

where y^{iv} is a constant (not zero).

Now, we define an error function, e

$$e = s - y$$

Subtracting Equations (8), (9) and (10) from (3), (4) and (5), respectively, we obtain

$$e_j = e_i + d_i e'_i + d_i^2 e''_i/2 + d_i^3 e'''_i/6 - d_i^4 y^{iv}/24 \qquad (11)$$

$$e'_j = e'_i + d_i e''_i + d_i^2 e''_i/2 - d_i^3 y^{iv}/6 \qquad (12)$$

$$e''_j = e''_i + d_i e'''_i - d_i^2 y^{iv}/2 \qquad (13)$$

Now we define a function, r, by

$$r \equiv e/y^{iv} \equiv (s - y)/y^{iv}$$

and divide equation by y^{iv} (not zero) to obtain (after transposition)

$$r_i - r_j + d_i r'_i + d_i^2 r''_i/2 + d_i^3 r'''_i/6 = d_i^4/24 \qquad (14)$$

$$r'_i - r'_j + d_i r''_i + d_i^2 r'''_i/2 = d_i^3/6 \qquad (15)$$

$$r''_i - r''_j + d_i r'''_i = d_i^2/2 \qquad (16)$$

We solve Equation (13) for r'''_i

$$r'''_i = (r''_j - r''_i)/d_i + d_i/2$$

and substitute in Equation (11) and (12) to obtain

$$r_i - r_j + d_i r'_i + d_i^2 r''_i/3 + d_i^2 r''_j/6 = - d_i^4/24 \qquad (17)$$

$$r'_i - r'_j + d_i r''_i/2 + d_i r''_j/2 = -d_i^3/12 \qquad (18)$$

Now we note that the coefficients for Equation (17) and (18) are

precisely those of Equations (6) and (7), however, unlike the latter, the

right hand sides of the former are not zero.

Where a value of y_k, y'_k or y''_k is specified, we have the corresponding

e_k, e'_k or e''_k equal zero, hence the corresponding of r_k, r'_k or r''_k equal zero.

Thus, transposing known terms to the right will not change the right

hand sides of (17) and (18). Further, the coefficients for the unknown

terms will be precisely those of Equation (8). Again we solve the

linear system

$$\vec{Ar} = \vec{d}$$

where A is the same matrix mentioned earlier, $\vec{r}$ represents $m - 2$ unknowns corresponding to the unknown r_k, r_k', r_k'' and $\vec{d}$ is a known vector constructed from the right hand sides of Equations (17) and (18). Note that we can solve for all the unknown r_k, r_k' and r_k'' and if y is indeed a quartic, we have

$$e_k = r_k \, y^{iv}$$
$$e_k' = r_k' \, y^{iv} \qquad \qquad \text{for } k = 1 \text{ to } m.$$
$$e_k'' = r_k'' \, y^{iv}$$

Thus, for a quartic, we can compute the errors of approximation of s, s' and s" at every knot.

In general, we are interested in the magnitude of the error rather than its sign, hence for quartic, y, we have

$$|e_k| = |r_k| \, |y^{iv}|$$
$$|e_k'| \quad |r_k'| \, |y^{iv}|$$
$$|e_k''| = |r_k''| \, |y^{iv}|$$

If y is not a quartic, y^{iv} is not constant, but if higher order derivatives are neglected, we can let

$$\bar{y}^{iv} = \max \, |y^{iv}|$$

and say

$$|e_k| \text{ is of the order } |r_k| \, \bar{y}^{iv}$$

$|e_k'|$ is of the order $|r_k'|$ $\bar{y}^{-iv}$

$|e_k''|$ is of the order $|r_k''|$ $\bar{y}^{-iv}$

Methods for estimating the value of $\bar{y}^{-iv}$ are left to the reader. The following are suggested

(1) $\bar{y}^{-iv} = \max |s_j''' - s_i'''|/d_i$ for $i = 1$ to $m - 2$

(2) Construct a polynomial of degree $m + 1$ (at least 4 for $m \geq 3$) from the $m + 2$ specified values (one of which must be a function value) and find the maximum absolute value for its fourth derivative.

INTERPOLATION

After s_i, s_i', s_i'' and s_i''' are determined for $i = 1$ to $m - 1$, we can always interpolate for s, s' and s'' for any x in $[x_1, x_m]$. First, we find the subinterval, $[x_i, x_j]$ in which x lies. Then we define

$$h = x - x_i$$

and use cubic expansions

$$s = s_i + hs_i' + h^2 s_i''/2 + h^3 s_i'''/6$$

$$s' = s_i' + hs_i'' + h^2 s_i'''/2$$

$$s'' = s_i'' + hs_i'''$$

If s is being used to approximate a function, y, and the error of approximation e is acceptable, for any x in $[x_1, x_m]$ we can let

$$y = s$$

If e' is acceptable, let

$$y' = s'$$

and if e" is acceptable, let

$$y'' = s''$$

COMPUTER CODE

A computer routine, UNVSPY, has been written in FORTRAN to compute all the unknown values for any properly specified cubic spline. If desired, it will also compute absolute error coefficients, $|r_k|$, $|r_k'|$ and $|r_k''|$ applicable for approximating some other function, y. It makes no estimate of y^{iv} and does not decide if the approximation is acceptable. However, if the user is able to do this for his particular function, y, and so desires, the subroutine on another option will interpolate for an array of x values the corresponding values, s, s' and s", which can be used to approximate y, y' and y" at these x values. Extrapolation is not permitted, all the values x must lie in $[x_1, x_m]$.

Listing and instructions for use of this routine are available from the author.

REFERENCES

1. Young, J., "Numerical Applications of Cubic Spline Functions," The
   Logistics Review, 3 (1967), pp. 9-14.

2. Young, J., "The Space of Cubic Splines with Specified Knots", The Logistics
   Review, 7 (1971), pp. 3-8.