A CONSISTENCY TEST FOR INTERPOLATORS

A. C. R. NEWBERY

Abstract.

The concept of "accuracy" is not satisfactory as a criterion for judging the quality of an interpolator. Accuracy in a given case is usually taken to mean proximity to a specific test function. However, infinitely many test functions could give rise to the same test data; proximity to one of these is not necessarily a sign of quality. We suggest that "consistency" is a more useful concept than accuracy in this context, and we demonstrate its usefulness.

Whenever an interpolation is performed on empirical data, one is implicitly making an assumption that the data could have been generated by a member of some family of functions, e.g. polynomials or piecewise cubics. On occasions there may be a physical motivation for preferring one family to another, but in the absence of such motivation one may be at a loss to decide which of various interpolation algorithms should be preferred. This article will define a decision-making process which uses exclusively information contained in the data set itself. It assumes no knowledge whatever concerning the analytic nature of the underlying function. In this context the concept of "accuracy" is of no use to us since the given data set could have been generated by infinitely many source functions, none of which is any more "right" than the others. Our proposal is to replace the "accuracy" concept by a "consistency" concept as described below.

We argue that the function \( f(x) \) is supposedly representable by \( n \) data points arranged in some pattern (e.g. equispaced or Chebyshev nodes) in \([a, b]\). That being so, for \( m > n \) it should also be adequately representable, and for \( m = n - 1 \) it should be representable without catastrophic loss of information (unless \( n \) is very small). Our enquiry is focussed on the case \( n = 9, m = n \pm 1 \) with equispaced nodes in \([-1, 1]\). We did a few spot tests in other circumstances in order to verify that conclusions drawn for larger \( n \) and/or for Chebyshev extrema would not be qualitatively different. Initially we shall assume that the interpolation method obeys the rule for linear operators, namely

\[
I(f(x) + g(x)) = I(f(x)) + I(g(x)),
\]

where \( I(f(x)) \) denotes the interpolant on \( f(x) \) based on a specific node set. The rule holds for splines and for algebraic and trigonometric polynomial interpolation.

Our test is as follows: (i) Define a node set \( \{x\} \) consisting of \( n(=9) \) equispaced points in \([-1, 1]\). Similarly define a second node set \( \{x'\} \) of \( m(=n \pm 1) \) equispaced points...
points in \([-1, 1]\), so that the two node sets have the two end-points in common, but no other common points. (ii) A function (to be specified later) is evaluated at \(\{x\}\); let the ordinates be called \(f_i\); the interpolant is formed and is evaluated at the node set \(\{x'\}\). (iii) The \(m\)-point interpolant based on the data set at \(\{x'\}\) is formed and is evaluated at \(\{x\}\). Let the ordinates be called \(\tilde{f}_i\). Ideally we would have \(\tilde{f}_i = f_i\) for all possible data sets \((x_i, f_i)\). If the interpolant satisfies (1), then the mapping from \(f\) to \(\tilde{f}\) is linear, i.e. \(\tilde{f} = Mf\) for some matrix \(M\), (whose eigenvalues ideally should all equal 1). We shall determine \(M\) explicitly thus; (iv) let \(f = e_1, e_2, \ldots, e_n\) in turn, where \(e_i\) are Euclidean basis vectors; hence the successive mappings \(Me_i\) are columns of \(M\). (v) Calculate the eigenvalues and vectors of \(M\). If \(M\) has an eigenvalue differing substantially from 1, then the associated eigenvector indicates an ordinate set for which the interpolation method is particularly likely to introduce distortion.

The interpolation methods satisfying (1) that we selected for study were (a) polynomial, (b) natural cubic spline, (c) "not-a-knot" cubic spline [2], (with third-derivative continuity at the second and penultimate nodes), (d) A local cubic method with first-derivative continuity, where first derivatives are parabolically approximated at the nodes [5, 3.8], (e) Trigonometric interpolation with an assumed period of 4 units. In the case \(m = n - 1\) it was found that there was always a zero eigenvalue, reflecting the fact that one cannot reduce the number of points from 9 to 8 without incurring a risk of losing all of the information contained in the 9 points. The associated eigenvector indicates the data set for which this will happen. In the case of the two splines it is reassuring to note in Table 1 that this data set is very conspicuous by its central peak, and all practitioners would have immediately warned against using a spline on such a data set. In the case \(m = n + 1\) the algebraic and trigonometric polynomials get perfect scores of all unit eigenvalues, as is obvious for theoretical reasons, but it is less reassuring to note that the splines miss perfection by a substantial margin. All the eigenvalues we ever encountered were real and in the range \([0, 1]\). We see no reason why this would necessarily have to be the case. In Table 1 we give partial results.

It can be shown that whenever the data abscissas are symmetrically located about the origin, the eigenvectors \(V\) will all have the palindromic property that \(V_i = V_{n+1-i}\) or alternatively \(V_i = -V_{n+1-i}\). In Table 1 we print only the two smallest eigenvalues, since these are the most important; we print the eigenvector associated with the smallest (non-unit) eigenvalue, suppressing the last four elements since these are implied by the palindromic property.

In the event that an interpolation method does not satisfy (1), we cannot perform the eigenanalysis above because the mappings are nonlinear. Methods in this class include rational interpolation, Prony's method [3] and Akima's method [1]. In these cases, however, we can still usefully apply the consistency concept on a case-by-case basis. Given a specific problem in the form of a data set, and assuming no knowledge about the source function, we can map the \(n\) points down