Freestyle Data Fitting and Global Temperatures

The method described here separates signal (trend) from noise in a set of measured bivariate data when there is no mathematical model for that signal. A computer program called `spline2` implements the algorithm, which the authors apply to laboratory and real-world example problems.

A previous series of articles[1–4] discussed several methods for fitting mathematical functions to data containing random errors, showing how to extract useful information from the results. Popular choices for the mathematical functions are polynomials and goniometric functions, as well as exponentials, Gaussians, Lorentzians, and various combinations of these. The data's overall shape often clues us in to what type of function seems to be appropriate, but what if we can't guess the mathematical function? Or, what if we don't want to rely on a possibly biased preconception about the type of function that's supposed to fit the data?

Problems of this kind happen frequently: scientists and engineers often confront data for which there is no theoretical model, for which it isn't clear, for example, whether the data contain a periodic component, or for which the shape's complexity appears to rule out any members of well-known function families. Yet researchers still want to separate random errors from the trend and find the analytic function that represents this trend as truthfully as possible. This article explains what to do in such circumstances. We present the algorithm for a computer program called `spline2`, which was specifically designed for this type of freestyle data fitting. An earlier version of the program appears elsewhere.[5] You can download `spline2` from the "Free Software" page at http://structureandchange.3me.tudelft.nl and an executable for Microsoft Windows from the "Data Analysis" page at www.geocities.com/karolewski, but it's also part of Macintosh's freeware program, Plot (http://plot.micw.eu).

A Model Problem

We can state the overall problem as follows: given N data points x_i, y_i ($i = 1, 2, \ldots, N$), of which the values y_i contain random errors ε_i,

$$y_i = f(x_i) + \varepsilon_i, \tag{1}$$

we want to find the best estimate of the function $y = f(x)$ without a priori knowledge about it. We assume the random error in y_i has the statistical properties

$$E(\varepsilon_i) = 0, \tag{2}$$

1521-9615/08/$25.00 © 2008 IEEE
Copublished by the IEEE CS and the AIP

Barend J. Thijsse
Delft University of Technology
Bert W. Rust
US National Institute of Standards and Technology

$$E(\varepsilon_i^2) = \sigma_i^2\,, \tag{3}$$

where E denotes expectation. The variance σ_i^2 of the random error in y_i can, in principle, differ for each data point, but in many cases, the variances are poorly known, so researchers often assume a common value for all σ_i^2. In this article, we work with normalized errors instead of with the errors themselves. The normalized error δ_i is

$$\delta_i = \frac{\varepsilon_i}{\sigma_i}\,. \tag{4}$$

We can consider the noise present in many data to be "white," which expresses the fact that the random errors in the data are independent of each other,

$$E(\delta_i\delta_j) = 0,\ (i \neq j). \tag{5}$$

However, we can't assume that this is always true. In many cases, often without the user knowing it, some kind of filtering or averaging process has acted on the data after the noise originated. The effect is that the random errors in neighboring data points become correlated—that is, Equation 5 is no longer true for $|i - j|$ equal to some small number. If we weren't aware of this—and it's easily overlooked in a visual inspection of the data—freestyle curve fitting could lead to very incorrect results. Keeping the quantity $E(\delta_i\delta_{i+n})$ at the correct value is a crucial part of the `spline2` algorithm, as we will see.

In this article, we assume that the autocorrelation function is of the following type:

$$\frac{E(\delta_i\delta_{i+n})}{E(\delta_i^2)} = \exp(-(x_{i+n} - x_i)/\xi)\,,\ (n \geq 0). \tag{6}$$

This exponentially decaying form is a good approximation for many practical cases. The quantity ξ is the autocorrelation length (for white noise, $\xi = 0$); in addition to providing the best estimate of $f(x)$, `spline2`'s algorithm also determines the value of ξ that agrees best with the data. To capture $f(x)$ in an analytic form, `spline2` uses flexible spline functions because they can represent virtually any type of trend. We briefly describe splines here, but they're fully discussed elsewhere.[6]

A spline, denoted here as $S(x)$, is a piecewise polynomial function. To visualize this, imagine that the data range $[x_1 \ldots x_N]$ is divided up into a certain number of intervals. The breakpoints between the intervals are called *interior knots*; the outer data points are also knots. Each polynomial piece of the spline covers an interval (here, we limit ourselves to the case in which all pieces are third-degree polynomials). The polynomial pieces are constructed in such a way that at each interior knot, the polynomial pieces to the left and to the right of the knot have the same value, the same first derivative, and the same second derivative. Only the highest derivative, a constant, is allowed to differ on the two sides of a knot. Taken together, these polynomial pieces form a smooth curve, the flexibility of which depends highly on the number of knots and their distribution along the x-axis. `spline2`'s principal task is to determine the optimal number and distribution of knots for approximating the trend in the data under consideration. The "Further Information" sidebar on p. 58 gives a brief review of existing literature about the smooth representation of data via splines.

Numerical Methods

We can summarize the method for obtaining the best approximation of $f(x)$ as follows: `spline2` systematically generates a collection of different trial splines $S(x)$—that is, splines with different knots—and fits them to the data in the least-squares sense. For each spline, `spline2` then calculates the set of normalized residuals $d_i (i = 1, 2, ..., N)$:

$$d_i = \frac{y_i - S(x_i)}{s_i}\,, \tag{7}$$

where the positive number s_i is a user-supplied estimate of the uncertainty in the ith point, so

$$s_i \text{ is the user estimate of } \sigma_i. \tag{8}$$

After a careful analysis of the sets of residuals, `spline2` ultimately decides for which of the trial cases the residuals' statistical properties are most consistent with those of δ_i in Equation 6. This indicates that the spline in question is the one that resembles $f(x)$ as closely as possible in a manner consistent with the postulated error structure. An important additional criterion is that given the choice between statistically equivalent candidate splines, the program will select the simplest spline—that is, the one with the fewest knots. This implies that Nature isn't expected to act in a more complicated way than necessary, but we won't discuss this principle of parsimony here.

Statistical Tests

Before we describe the fitting algorithm in more detail, let's first look at the kind of statistical test to apply in order to decide if a particular function $S(x)$ fits the data well. In doing so, we should also include the cases in which the data points' error variances are imprecisely known. Normal practice after performing a least-squares fit—that is, after varying the mathematical function's adjustable parameters until the residual variance is at a minimum—is to apply a χ^2-test to it. We calculate the statistic

$$r^2 \equiv \frac{\sum_{i=1}^{N} \left(\frac{y_i - S(x_i)}{s_i} \right)^2}{N-m} = \frac{\sum_{i=1}^{N} d_i^2}{N-m}, \tag{9}$$

where m is the number of adjustable parameters. If $S(x)$ is the correct fitting function, and if the values s_i are the correct estimates of σ_i, this statistic is χ^2-distributed, with an expected value 1 and standard deviation $\sqrt{2/(N-m)}$. Thus, the value of r^2 actually found is easy to test against the interval $1 \pm \sqrt{2/(N-m)}$.

If, however, our estimates of the error variances are incorrect, the χ^2-test quickly loses its power, and the outcome ceases to accurately impute the correctness of the function fitted to the data at hand. Suppose, for example, that we underestimate all uncertainties σ_i by 10 percent and therefore all variances by roughly 20 percent. This misjudgment would be small, but the effect will be that the correct $S(x)$ gives rise to an r^2 that's too large by a factor 1.20. Therefore, a potentially correct r^2 (= 1) goes outside the interval of one standard deviation for $N - m$ as low as 50, which is a common situation for data fitting. Thus, the χ^2-test can be useless because it's too critically dependent on our precise knowledge of data errors. (A common, but dangerous, practice in such cases is to invert the reasoning by picking a fitting function and assuming that the fit is correct; we then use the value found for r^2 to estimate the uncertainties through $\sigma_i = rs_i$.)

Clearly, we need a better test. We can't hope to achieve complete immunity to all types of misjudgment, but much can be gained if we could use a statistic that makes us at least less sensitive to such errors. This is especially important for the present case of spline fitting because we can give enormous (even too much) flexibility to the spline by increasing the number of knots and varying their distribution. It's therefore vital to prevent splines from using this flexibility to overfit the noise. The Durbin-Watson test[7–9] gives us the desired insensitivity, and it's with this test that we can begin to explain the algorithm by which `spline2` finds the best-fitting spline. To incorporate the effects of autocorrelation, we modified the Durbin-Watson statistic somewhat.

The Durbin-Watson Test

In its original form, the Durbin-Watson test is applied to a statistic that we call $Q(1, 0)$ in this article. We define it as

$$Q(1,0) \equiv \frac{N-1}{N} \frac{\left\langle (d_{i+1} - d_i)^2 \right\rangle_{(1)}}{\left\langle d_i^2 \right\rangle}, \tag{10}$$

where we used the following notation for averages:

$$\langle \ldots \rangle \equiv \frac{1}{N} \sum_{i=1}^{N} (\ldots), \tag{11}$$

$$\langle \ldots \rangle_{(n)} \equiv \frac{1}{N-n} \sum_{i=1}^{N-n} (\ldots). \tag{12}$$

To calculate $Q(1, 0)$ after we've determined the least-squares fit of $S(x)$ to the data, we need the values d_i, which depend on the estimates s_i of the data uncertainties (Equation 7). If we know nothing about the size of these uncertainties, we can't do much better than set all s_i equal to 1; if we have at least some notion of which data are more reliable and which are less, we should try to give each s_i a value proportional to the true uncertainty. The crucial point here is that agreement between s_i and σ_i in an absolute sense isn't necessary—agreement on a relative scale suffices. The reason for this is that $Q(1, 0)$ is a ratio of two estimates of the residual variance: one based on the magnitudes of the point residuals, and the other based on the serial correlation between them. Because it's a ratio, $Q(1, 0)$ is unaffected by any common factor by which the weighted residuals might be incorrect. Problems like the 20 percent underestimation in the example of the χ^2-test therefore don't occur if we use $Q(1, 0)$. This is the Durbin-Watson test's great advantage: the outcome is much less sensitive to user misjudgments of noise magnitude. As we'll see, this is also true for misjudgments other than a simple common factor.

The statistical properties of $Q(1, 0)$ are well known.[7–9] For correct approximants, the expected value of $Q(1, 0)$ is approximately $2(N - 1)/N$, and the standard deviation is roughly $2/\sqrt{N-m}$, where, in the present context of spline fitting, $m = L + D$, with L representing the number of intervals and D the spline's polynomial degree. The exact values of the expected value and the standard deviation depend on the x_i and on the ap-

I. For a range of ξ values, from $\xi = 0$ to $\xi = 3\langle\Delta x\rangle$—in increments of $\langle\Delta x\rangle/10$ until $\xi = \langle\Delta x\rangle$, and in increments of $\langle\Delta x\rangle/5$ thereafter—`spline2` executes the following steps:[5]

1. A series of trial "equal information" splines $S_L(x)$ with gradually increasing numbers of intervals L is least-squares fitted to the data. The term "equal information" refers to splines in which the knots are positioned in such a way that each interval contains the same number of data points (or as close to this as possible). The series starts with $L = 1$, and at every step L is increased by 1 or by 5 percent, whichever is larger. For each fit, q is calculated.
2. As soon as the tolerant Durbin-Watson test applied to q indicates that the fit is acceptable, this fit's spline is considered a good first approximation. We denote this spline as $S_{L_1}(x)$.
3. A second series of trial splines is fitted, starting from $S_{L_1}(x)$, but this time L is decreased at each step (by 1 or by 5 percent, whichever is larger); simultaneously, an algorithm is applied that optimizes the distribution of knots on the basis of the preceding spline fit.[6]
4. The strict Durbin-Watson test is applied to all spline fits of this series. Of the acceptable ones, the spline with the smallest number of knots is considered optimal for the particular value of ξ under investigation. Special case: if none of the splines in this series is found acceptable, the spline series of step 1 is extended by one more spline (L_1 is increased), and the process is re-entered at step 3.
5. A parameter α is calculated for the optimal spline found in step 4. This parameter measures how closely the spline fit's residuals conform to the assumed autocorrelation function. Experimentation has shown that the following expression performs well:

$$\alpha = \left|\min_n C(n,\xi)\right| + \left|\max_n C(n,\xi)\right|,$$

where all n values in the range [1 ... $n_{\max}$] are considered (see Equation 19 in the main text). Note that the value of α depends on ξ.

II. After the loop over ξ is completed, all values α are examined. The smallest of these represents the best correspondence with the assumed autocorrelation function. The ξ value that gave rise to this smallest α is then selected as the most probable value, and the corresponding optimal spline is finally presented to the user as the overall best. As output, `spline2` also produces the estimate r of the true data uncertainties (relative to the user-supplied values). By rewriting Equation 9 in the main text, we see that

$$r = \left[\sum_{i=1}^{N} w_i \left(\frac{\sigma_i}{s_i}\right)^2\right]^{1/2},$$

where the weight factor w_i is given by

$$w_i = \frac{1}{N-m}\left(\frac{y_i - S(x_i)}{\sigma_i}\right)^2.$$

The individual values w_i are unknown, but we do know that $E(\Sigma_{i=1}^{N} w_i) = 1$. We can therefore come to the conclusion that $\bar{\sigma} = r\bar{s}$, where the bars indicate averaging. This equation says that the best spline fit indicates that the uncertainties in the data are r times as large as the user estimates.

Figure 1. The spline fit algorithm. In the main text, the optimal spline fits are characterized by the parameters L, r, α, and ξ.

proximating function, but we can base statistical tests on two limiting distributions of $Q(1, 0)$. We normally apply these *tolerant* and *strict* Durbin-Watson tests with a confidence level of 0.95. Incidentally, the application of these tests as used in `spline2` is less straightforward than it seems because the program uses a more generalized version of the statistic than $Q(1, 0)$—namely, one that takes into account autocorrelation effects. To see how autocorrelation comes in, we first rewrite Equation 10 as

$$Q(1,0) \equiv \frac{N-1}{N}\left(\frac{\left\langle d_{i+1}^2\right\rangle_{(1)} + \left\langle d_i^2\right\rangle_{(1)}}{\left\langle d_i^2\right\rangle} - 2C(1,0)\right), \qquad (13)$$

with

$$C(1,0) \equiv \frac{\left\langle d_{i+1} d_i\right\rangle_{(1)}}{\left\langle d_i^2\right\rangle}. \qquad (14)$$

For uncorrelated data, the expected value of $C(1, 0)$ is zero, which is a statement similar to Equation 5. In fact, for any n, the expected value of $C(n, 0)$, defined as

$$C(n,0) \equiv \frac{\left\langle d_{i+n} d_i\right\rangle_{(n)}}{\left\langle d_i^2\right\rangle}, \qquad (15)$$

should be zero for truly white noise. If this quantity is adjusted by the form of the expected correlation given by Equation 6,

$$C(n,\xi) \equiv \frac{\langle d_{i+n} d_i \rangle_{(n)}}{\langle d_i^2 \rangle} - \langle \exp(-(x_{i+n} - x_i)/\xi) \rangle_{(n)} . \quad (16)$$

This statistic, again, is expected to be zero for a good fit (of course, any other assumed autocorrelation function could replace the exponential function in Equation 16). Inserting this expression into Equation 13 we find a modified Durbin-Watson statistic

$$Q(n,\xi) \equiv \frac{N-1}{N}\left(\frac{\langle d_{i+n}^2 \rangle_{(n)} + \langle d_i^2 \rangle_{(n)}}{\langle d_i^2 \rangle} - 2C(n,\xi) \right). \quad (17)$$

We still must take one final step. The quantity $Q(n, \xi)$ refers to one n value only—that is, we compare the actual autocorrelation function with the assumed autocorrelation function on the basis of only one value for the data index spacing n. To obtain a comparison over a range of spacings, we take the average over several n values and arrive at our final generalized Durbin-Watson statistic q,

$$q \equiv \frac{1}{n_{\max}} \sum_{n=1}^{n_{\max}} Q(n,\xi) , \quad (18)$$

which is a function of ξ. In the `spline2` algorithm, the value of $n_{\max}$ is large enough to let n cover the most significant support of the nonzero part of the autocorrelation function,

$$n_{\max} = \operatorname{int}\left(3 \frac{\xi}{\langle \Delta x \rangle} + 3 \right), \quad (19)$$

where the average data spacing $\langle \Delta x \rangle$ is given by

$$\langle \Delta x \rangle \equiv \frac{x_N - x_1}{N-1} . \quad (20)$$

This completes the definition of q, the pivotal quantity that's Durbin-Watson tested in the construction of the best spline approximation to noisy data. Figure 1 gives the algorithm's full details.

Application to Pulse-Counting Data

As a first illustration of the fitting method proposed here, Figure 2 shows spectrometer data (N = 536, $\langle \Delta x \rangle$ = 3.16) obtained in a gas-desorption experiment.[5] Because the data y_i are pulse counts and should obey Poisson statistics, we know the noise component's standard deviation. Therefore, we have set the values s_i equal to $\sqrt{y_i}$. Figure 1 shows the best cubic spline approximation the algorithm found, $S(x)$, together with the data. To highlight the curve's detailed structure, $S(x)$ is also shown shifted upward. The numerical results are L = 34, ξ = 0.95, α = 0.045, and r = 0.975, with a Durbin-Watson statistic q = 2.05.

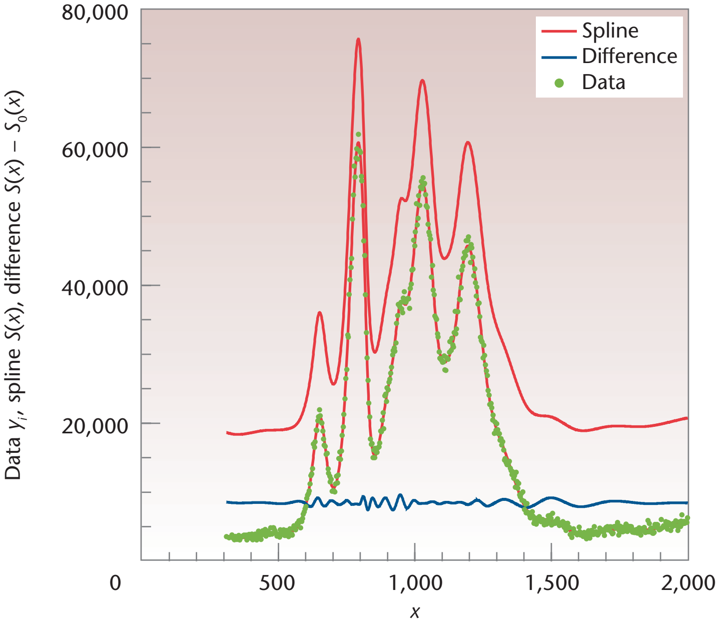

Figure 2. Pulse-counting data and fitted spline $S(x)$. To highlight the curve's detailed structure, $S(x)$ is also shown shifted upward. The curve marked "Difference" is $S(x) - S_0(x)$, where $S_0(x)$ is the fit found under the assumption that all data would have equal uncertainties.

We can see that the spline successfully captures all the essential trends in the data, and that it requires 34 intervals (33 internal knots) to do so. Also, the spline's first and second derivatives (not shown) have very acceptable forms, exhibiting none of the wild oscillations that would indicate overfitting. The value of r, nearly equal to 1, confirms that we estimated the data uncertainties correctly. The correlation length ξ is found to be much smaller than the average data spacing (only 30 percent), which means that the autocorrelation function's value between neighboring points is only exp(–1/0.3) = 0.036. This is so close to zero that we could consider the data to be effectively uncorrelated. In fact, this is in accordance with the experimental situation—the very small value of α confirms this. If, on the other hand, we pretend to know nothing about the data uncertainties and set all s_i equal to 1, the result is a spline $S_0(x)$ with the following statistics: L = 30, ξ = 1.58, α = 0.116, and r = 816. This spline is virtually identical to the previous one, as exemplified by the fact that the largest difference in peak height is only 353 vertical units, and the largest

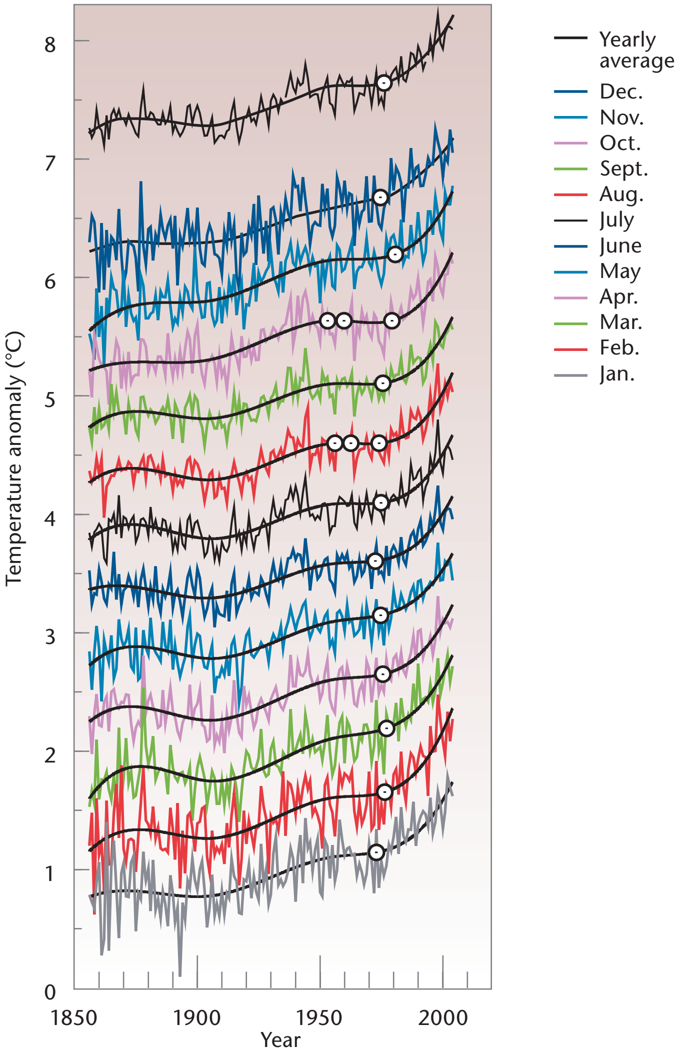

Figure 3. Global temperature anomalies from 1856 to 2004. We plotted January to December data from bottom to top (vertically shifted according to their average values), whereas the yearly averages are shown as the topmost data set. The smooth curves are the fitted splines $S(x)$. Open symbols mark the points where $S(x) = 0$, thus defining the positions of the data sets along the vertical axis.

difference in peak position is only 3.0 horizontal units. Figure 2 shows the difference between $S(x)$ and $S_0(x)$. This test illustrates that the fitting algorithm is robust against misjudgments of data uncertainties.

Application to Global Temperatures

To maintain continuity with previous articles,[1–4] we next apply the freestyle fitting method to global temperature data (see www.cru.uea.ac.uk/cru/data/temperature/#datdow), which are updated versions of the ones analyzed in the previous articles. Figure 3 shows the global monthly and yearly temperature anomalies from 1856 to 2004. (Temperature anomalies are the mean global temperature differences with respect to the 1961 to 1990 average.) We plotted January to December data from bottom to top, whereas the yearly averages are shown as the topmost data set. All data sets have $N = 149$, $\langle \Delta x \rangle = 1$. The depicted curves are the spline fits found by `spline2`. We discuss the yearly data first.

Yearly Temperature Averages

For yearly average temperatures, `spline2` produces the three-interval spline fit shown as the topmost curve in Figure 3. Note that we have set all s_i values equal to 1. The spline statistics are $L = 3$, $\xi = 0.7$, $\alpha = 0.142$, and $r = 0.100$, which tells us that the random fluctuations in the yearly average temperature have a standard deviation of 0.100 degrees Centigrade and that the autocorrelation length—or, rather, autocorrelation time—is 0.7 years. The small value of α shows that the correlations agree well with the assumed exponential form in Equation 6; Figure 4 shows this in more detail. Further inspection shows that α has indeed reached a pronounced minimum for $\xi = 0.7$—for example, $\alpha = 0.306$ for $\xi = 0.3$ and $\alpha = 0.224$ for $\xi = 1.1$. If no correlation at all is assumed ($\xi = 0$), α becomes as high as 0.342. We conclude that the autocorrelation time of 0.7 years is a real phenomenon. For completeness, we mention that the standard deviation $r = 0.100$ lies between the r values produced by ordinary 6th and 7th degree polynomial fits over the entire time range.

Individual Months

To study global temperature data in more detail, we look next at the data for individual months, where again we have set all values s_i equal to 1. Reverting to Figure 3, we see that all months essentially exhibit the same behavior. This is even clearer in Figure 5 on p. 56, which shows the splines' derivatives in the same arrangement as in Figure 3. The closed symbols in Figure 5 denote the minima and maxima of the derivatives, marking the points at which climatic trends begin to change. We can see that for all months, these points group closely around the same years: 1889, 1931, and 1964 (vertical lines). The similarity of the monthly data is striking and seems to emphasize the fact that the global climate is a system with time memory, something already suggested by the autocorrelation time of 0.7 years found for the yearly averages. Additionally, the results indicate that external influences must be active, such as solar or human activities (combustion of fossil fuels), which are so strong that the global climate as a whole responds with the same trend.

In the next section, we study the relation between global temperature and atmospheric concentration of CO_2. The year 1964 emerges as significant: from this point onward, the temperature derivatives for all months increase continuously. What's more, over the past 10 years or so, the rates of temperature increase have been larger than ever (since 1856) and continue to increase, even up to the last year considered.

Figure 6 on p. 57 shows the estimates *r* of the standard deviation of the fluctuations for the January to December temperatures (open squares) together with the *r* value of the yearly averages (closed square). In the monthly data, we see distinct seasonal effects: *r* varies from 0.11 degrees (August and September) to 0.21 degrees (February). This is interesting in itself because global temperature data cover both the northern and southern hemispheres, and we wouldn't immediately expect to see a difference between summer and winter. Evidence also exists that the correlation effects in the full, continuous sequence of monthly temperature data are more complicated than can be explained by a simple exponential function. Figure 6 shows this as well, from the correlation times ξ between the same months in different years and the corresponding values of α. For comparison, we also included values for the yearly averages in the plot (closed symbols). What `spline2` finds is that the January, February, June, and November data appear to be uncorrelated from year to year, and that the correlation times for the other months fall between 0.3 and 0.8 years; the small values for α ($\approx$ 0.1) suggest that these are real correlation times.

We must make one additional remark. We didn't generate the spline fits to the 12 monthly data sets in Figure 3 with the exact algorithm described earlier. We set the value of L_1, the number of intervals needed for the least-squares spline to qualify as "good enough as a first approximation," to 3 instead of letting the tolerant Durbin-Watson test decide it. If the value assignment were left to the Durbin-Watson test, it would come up with the value $L_1 = 1$ for the winter months (November to April). In other words, for the months with the largest *r* values (see Figure 6), the random temperature fluctuations are apparently just large enough to let the algorithm decide that a single cubic polynomial, rather than a three-interval cubic spline, fits the data sufficiently well. This in itself isn't a serious matter, but we chose to force the algorithm to wait a little longer before invoking the knot-optimization: experience has shown that in cases in which the ratio between the noise amplitude and the "trend features magnitude" is high, it's usually better to let the knot-optimization routine start from a spline that already has a certain flexibility. After all, if the correct trend *is* a single polynomial, the algorithm will find it anyway. Setting L_1 to 3 therefore constitutes no illegitimate act of user interference.

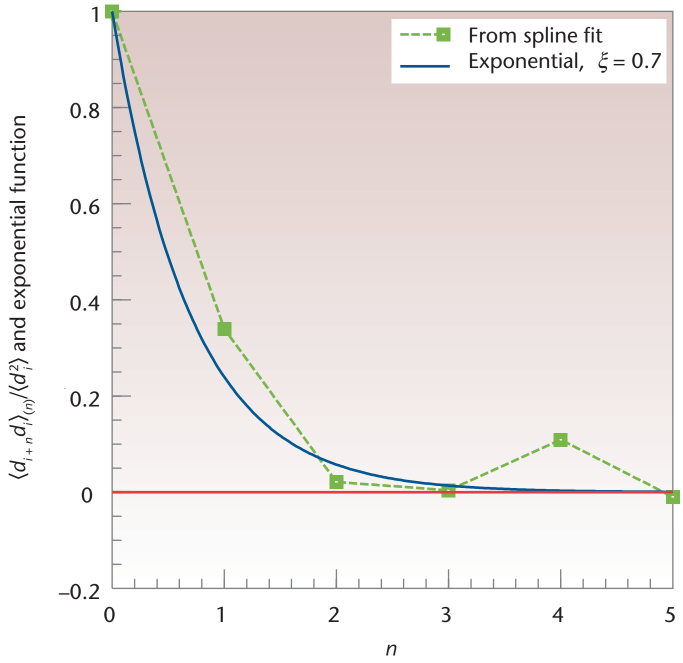

Figure 4. Noise autocorrelation function agreement. Points are for the spline fit of the uppermost data in Figure 3. The curve is an exponential function with $\xi = 0.7$.

As a second small deviation from the usual mode of operation, we applied the Durbin-Watson tests with a 97 percent confidence level instead of 95 percent. With 95 percent, the September spline turned out to be a two-interval spline, but using 97 percent had the effect that the September spline became a three-interval one, whereas the other splines remained unchanged. Given the arbitrariness of the confidence interval's value, this adaptation can't be seen as an essential interference. Overall, our conviction is that for all months, the three-interval splines should be considered best; the synchronous behavior in Figure 5 certainly emphasizes this outcome. Another sign that "fully automatic" splines are likely to be inferior to three-interval ones is that the difference between the average of the monthly splines and the topmost spline in Figure 5 increases from 0.00013 to 0.08123 degrees if we choose the automatic splines.

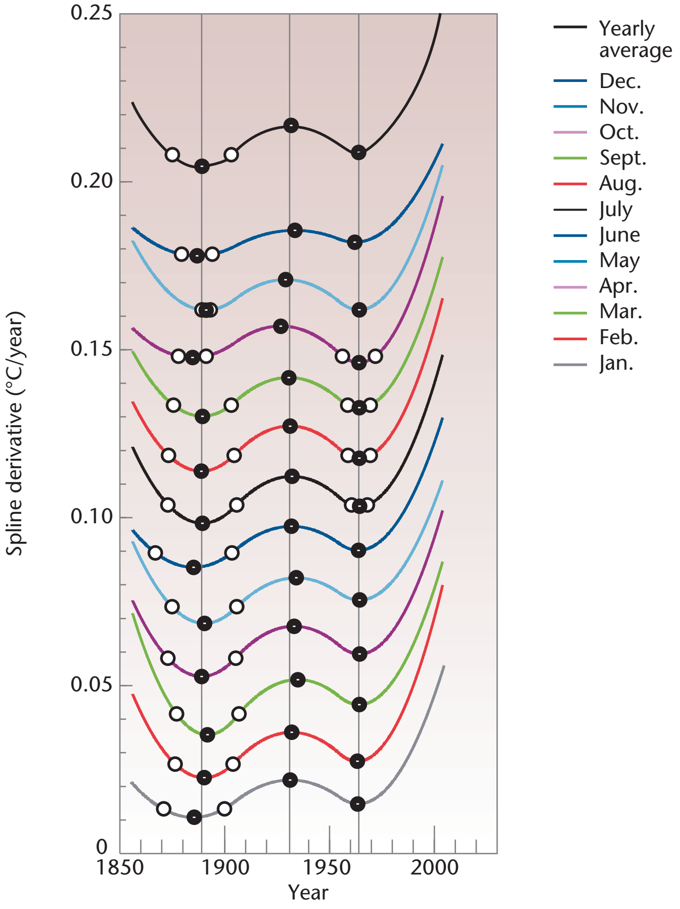

Figure 5. Derivatives of Figure 3's splines. Open symbols mark the points at which each derivative assumes the value 0, thus defining the positions of the curves along the vertical axis. Closed symbols mark the derivatives' minima and maxima, which is where climatic trends begin to change. Note that for all months, these points group closely around the same years: 1889, 1931, and 1964 (vertical lines).

CO_2 Concentrations vs. Temperature

As we just showed, a spline fit designed to separate signal from noise in a measured record provides a model-independent mathematical representation of the quantity or process being measured. This, in turn, can help us model some other related observed quantity or process. To illustrate this procedure, let's consider the relation between the global temperature variations just discussed and the atmospheric concentration of CO_2.

Figure 7 gives a plot of the atmospheric concentrations of CO_2 measured in Antarctica for the years 1647 to 2004.[10–12] Taking $t = 0$ at epoch 1856.0, we let $c(t)$ be the atmospheric concentration at time t. We use the average value for the years 1647 to 1764 as an estimate of the preindustrial concentration, and the optimal `spline2` fit to the years 1765 to 2004 as a mathematical representation for $c(t)$ in a model for the temperature anomalies. The `spline2` fit used five intervals and gave residuals with a root-mean-square error $r = 1.035$ parts per million by volume (ppmv).

To model the temperature variations, we let T_0 be the temperature anomaly corresponding to $c(t) = c_0$ and assumed a linear relation

$$\frac{dT}{dc} = \eta \Rightarrow T(t) = T_0 + \eta[c(t) - c_0], \tag{21}$$

where the parameters T_0 and η are to be estimated by least-squares fitting. But this expression defines only a baseline for the temperature variations: to complete the model, we must add a sinusoid to account for the approximately 70-year oscillation that Michael Schlesinger and Navin Ramankutty first reported.[13] This cycle, which was very apparent in all of Figure 3's `spline2` plots, is thought to come from ocean–atmospheric interaction and hence is independent of CO_2 concentration. Adding it to our model gives

$$T(t) = T_0 + \eta[c(t) - c_0] + A \sin\left[\frac{2\pi}{\tau}(t + \phi)\right], \tag{22}$$

with free parameters T_0, η, A, ϕ, and τ. A nonlinear least-squares fit to the temperature data gives the parameter estimates

$$\hat{T}_0 = -0.507 \pm .016\,[°C]$$

$$\hat{\eta} = 0.01039 \pm .00042\,[°C / \text{ppmv}]$$

$$\hat{A} = 0.099 \pm .012\,[°C]$$

$$\hat{\tau} = 71.5 \pm 2.2\,[\text{yr}]$$

$$\hat{\phi} = -1.0 \pm 1.4\,[\text{yr}].$$

The sum of squared residuals and corrected total sum of squares are $SSR = 1.2674$ and $CTSS = 8.5563$, so the coefficient of determination is

$$R^2 = 1 - \frac{SSR}{CTSS} = 0.8519.$$

This means that the fit and the residuals explain 85.19 percent and 14.81 percent, respectively, of the data variance. We can also decompose the data into variance components:

$$T_{obs} \equiv \text{baseline} + \text{sinusoid} + \text{noise},$$

with

Baseline ≡ $\hat{T}_0 + \hat{\eta}[c(t) - c_0]$,

Sinusoid ≡ $\hat{A}\sin[\hat{\omega}(t + \hat{\phi})]$, and

Noise ≡ residuals for the fit.

An approximate analysis of variance shows that the baseline and sinusoid account for approximately 77 and 8 percent of the variance, respectively.

Figure 8 on p. 59 gives a plot of the fit, a plot of the `spline2` fit, and a plot of their difference. The good agreement between the two fits argues favorably for both approaches. The baseline, which is also plotted, suggests that the troposphere has warmed by approximately 0.9°C since 1856, that the warming is directly attributable to the increase in atmospheric CO_2 during that period, and that the warming is accelerating.

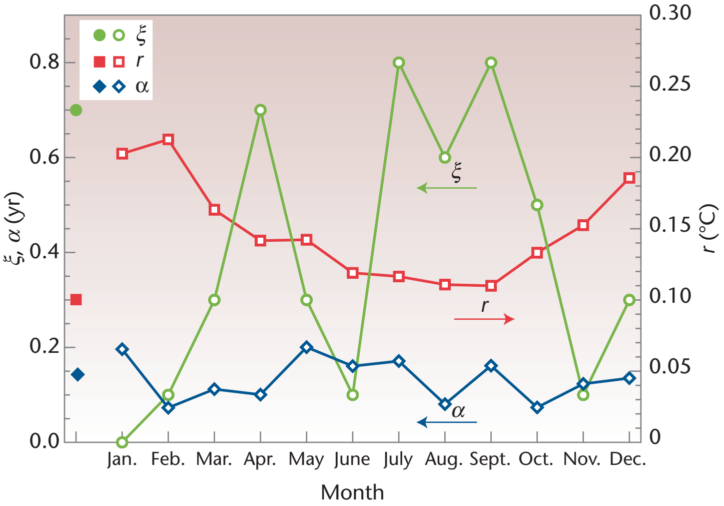

Figure 6. Parameters. For Figure 3's spline fits, *r* (squares), *ξ* (circles), and *α* (diamonds) versus month (J-D = Jan.-Dec.). The closed symbols (Yr) are the corresponding values for the yearly average temperatures.

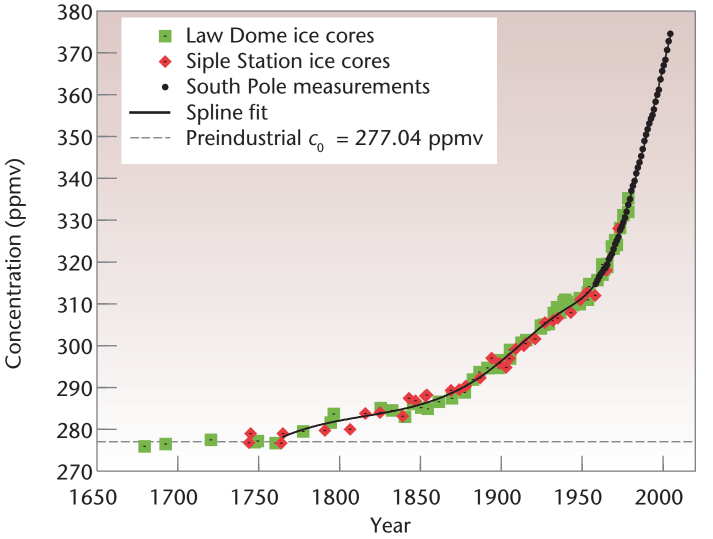

Figure 7. Atmospheric CO_2 concentrations measured in Antarctica. The units are parts per million by volume (ppmv), solid circles are atmospheric measurements made at the South Pole station,[10] open squares are proxy measurements taken from the Law Dome ice cores,[11] and diamonds are proxy measurements from the Siple Station ice core.[12] The horizontal dashed line is the average value for the years 1647 to 1764, and the solid curve is the optimal `spline2` fit to the data for the years 1765 to 2004. The measured data come from http://cdiac.ornl.gov/trends/co2/contents.htm.

We found over the years that the spline approximation method described in this article works surprisingly well in many cases—specifically, it provides researchers with a routine tool for analyzing noisy data, and common tasks such as interpolation and determining peak maxima, derivatives, and baselines no longer require fitting preselected mathematical functions. However, the method becomes less reliable if the noise's autocorrelation function varies significantly over the data range and if the signal contains sharp steps or cusps. We also found that weak periodic components in the signal are sometimes incompletely recognized; work is ongoing to improve this.

We've included the examples of global temperatures and CO_2 concentration as a contribution to the climate debate. Extrapolation of the observed trends is, of course, not possible using the methods presented here, but it *is* rather suggestive that a mathematical method fully ignorant of the global climate comes to the same conclusions as a CO_2-based model that includes ocean–atmospheric interactions. Work is under way to take the CO_2 analysis one step further: by using a `spline2` approximation to records of fossil-fuel emissions and land-use changes, we're developing a mathematical model that relates these human CO_2 production data to measured atmospheric CO_2 concentrations. This would be helpful in predicting future temperature scenarios.

Acknowledgments

We're much indebted to Mark Hollanders and Jan Hendrikse for essential contributions to `spline2`'s original algorithms; to Marcus Karolewski, presently at the University of Brunei Darussalam, for programming the Win32 wrapper `spline2x`; and to Michael

Further Information

Spline fitting was originally developed[1,2] for interpolating a set of data points $\{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$ with highly accurate y_i values. Each data point provided a knot for the spline function $S(x)$, which was constrained to reproduce the y_i values exactly—that is, $S(x_i) = y_i$. It wasn't long before people realized that an even more valuable tool would be splines that could smooth data sets in which the y_i values were corrupted by random measuring errors. These splines would be designed to approximate the y_i in such a way that measuring errors were relegated to the residuals $y_i - S(x_i)$. The two main approaches to this goal were *smoothing* splines[3,4] and *regression* splines, or least-squares splines.[5,6]

Smoothing splines—like interpolating splines—place a knot at every x_i, but they're designed not to satisfy $S(x_i) = y_i$, but to minimize

$$(1-q)\sum_{i=1}^{N}\left[\frac{y_i - S(x_i)}{\sigma_i}\right]^2 + q\int_{x_1}^{x_N}[S''(x)]^2 dx\,,$$

where σ_i^2 is some measure of the variance of the error in y_i, and q is chosen from the interval $0 < q < 1$ to adjust the amount of smoothing. The equation is a convex combination of a term that measures fidelity to the data and a term that measures the smoothness of $S(x)$. In the limiting case $q = 0$, $S(x)$ becomes the interpolating spline, and as $q \to 1$, $S(x)$ reduces to a simple straight line fit to the data. The interesting question is how to choose q to give an optimal separation of signal from noise. The best answer given so far is to choose q to minimize Grace Wahba's generalized cross-validation function.[7] More details appear in her classic book.[8]

Regression splines achieve smoothing by reducing the number of knots so that several data points are included in each interval between pairs of adjacent knots; the resulting splines are fit to the data set in the usual least-squares sense. In general, fewer knots give smoother splines, so the interesting questions become the number of knots to use and their placement in the interval $x_1 < x < x_N$ to produce the optimal separation of signal from noise. Researchers have developed several strategies for making these choices,[9–11] and one work in particular[12] gives the precursor to the strategy developed in the main text. Two other good sources of information about regression splines and smoothing splines appear elsewhere.[13,14]

References

1. I.J. Schoenberg, "Contributions to the Problem of Approximation of Equidistant Data by Analytic Functions," *Quarterly of Applied Mathematics*, vol. 4, no. 1, 1946, pp. 45–99, and no. 2, pp. 112–141.
2. I.J. Schoenberg, "Spline Interpolation and the Higher Derivatives," *Proc. Nat'l Academies Science USA*, vol. 51, no. 1, 1964, pp. 24–28.
3. C.H. Reinsch, "Smoothing by Spline Functions," *Numerische Mathematik*, vol. 10, no. 3, 1967, pp. 177–183.
4. C.H. Reinsch, "Smoothing by Spline Functions, II," *Numerische Mathematik*, vol. 16, no. 5, 1971, pp. 451–454.
5. C. de Boor and J.R. Rice, *Least Squares Cubic Spline Approximation I, Fixed Knots*, CSD TR 20, Dept. of Computer Science, Purdue Univ., Apr. 1968.
6. C. de Boor and J.R. Rice, *Least Squares Cubic Spline Approximation II, Variable Knots*, CSD TR 21, Dept. of Computer Science, Purdue Univ., Apr. 1968.
7. P. Craven and G. Wahba, "Smoothing Noisy Data with Spline Functions," *Numerische Mathematik*, vol. 31, no. 4, 1979, pp. 377–403.
8. G. Wahba, *Spline Models for Observational Data*, SIAM Press, 1990.
9. J.H. Friedman and B.W. Silverman, "Flexible Parsimonious Smoothing and Additive Modeling," *Technometrics*, vol. 31, no. 1, 1989, pp. 3–21.
10. H. Schwetlick and T. Schutze, "Least Squares Approximation by Splines with Free Knots," *BIT*, vol. 35, no. 3, 1995, pp. 361–384.
11. T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer, 2001.
12. B.J. Thijsse, M.A. Hollanders, and J. Hendrikse, "A Practical Algorithm for Least-Squares Spline Approximation of Data Containing Noise," *Computers in Physics*, vol. 12, no. 4, 1998, pp. 393–399.
13. C. de Boor, *A Practical Guide to Splines*, Springer, 1978.
14. R.L. Eubank, *Nonparametric Regression and Spline Smoothing*, Marcel Dekker, 1999.

Wesemann at the Fritz Haber Institute (Berlin) for including `spline2` in Plot. We also gratefully acknowledge David Gilsinn and Stefan Leigh at NIST for critically reading the manuscript.

References

1. B.W. Rust, "Fitting Nature's Basic Functions, Part I: Polynomials and Linear Least Squares," *Computing in Science & Eng.*, vol. 3, no. 5, 2001, pp. 84–89.
2. B.W. Rust, "Fitting Nature's Basic Functions, Part II: Estimating Uncertainties and Testing Hypotheses," *Computing in Science & Eng.*, vol. 3, no. 6, 2001, pp. 60–64.
3. B.W. Rust, "Fitting Nature's Basic Functions, Part III: Exponentials, Sinusoids, and Nonlinear Least Squares," *Computing in Science & Eng*, vol. 4, no. 4, 2002, pp. 72–77.
4. B.W. Rust, "Fitting Nature's Basic Functions, Part IV: The Variable Projection Algorithm," *Computing in Science & Eng.*, vol. 5, no. 2, 2003, pp. 74–79.
5. B.J. Thijsse, M.A. Hollanders, and J. Hendrikse, "A Practical Algorithm for Least-Squares Spline Approximation of Data Containing Noise," *Computers in Physics*, vol. 12, no. 4, 1998, pp. 393–399.
6. C. de Boor, *A Practical Guide to Splines*, Springer, 1978.
7. J. Durbin and G.S. Watson, "Testing for Serial Correlation in Least Squares Regression, I," *Biometrika*, vol. 37, nos. 3–4,

1950, pp. 409–428.

8. J. Durbin and G.S. Watson, “Testing for Serial Correlation in Least Squares Regression, II,” *Biometrika*, vol. 38, nos. 1–2, 1951, pp. 159–178.
9. J. Durbin and G.S. Watson, “Testing for Serial Correlation in Least Squares Regression, III,” *Biometrika*, vol. 58, no. 1, 1971, pp. 1–19.
10. C.D. Keeling and T.D. Whorf, “Atmospheric CO_2 records from Sites in the SIO Air Sampling Network,” *Trends: A Compendium of Data on Global Change*, Carbon Dioxide Information Analysis Ctr., 2005; http://cdiac.ornl.gov/ftp/trends/co2/sposio.co2.
11. D.M. Etheridge et al., “Historical CO_2 Records from the Law Dome DE08, DE08-2, and DSS Ice Cores,” *Trends: A Compendium of Data on Global Change*, Carbon Dioxide Information Analysis Ctr., 2005; http://cdiac.ornl.gov/ftp/trends/co2/lawdome.combined.dat.
12. A. Neftel et al., “Historical CO_2 Record from the Siple Station Ice Core,” *Trends: A Compendium of Data on Global Change*, Carbon Dioxide Information Analysis Ctr., 2005; http://cdiac.ornl.gov/ftp/trends/co2/siple2.013.
13. M.E. Schlesinger and N. Ramankutty, “An Oscillation in the Global Climate System of Period 65–70 Years,” *Nature*, vol. 367, Feb. 1994, pp. 723–726.

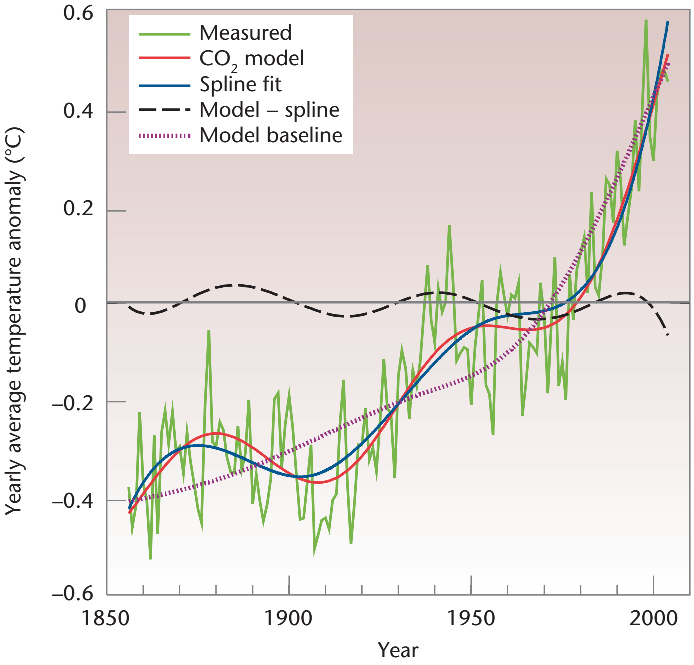

Figure 8. Fit comparison. Here, we compare the measured yearly global average temperature anomalies of Figure 3, both with the CO_2 model fit of Equation 22 and with the direct `spline2` fit. The dashed curve shows the difference between the model and the spline fit. The violet curve is the baseline $\hat{T}_0 + \hat{\eta}[c(t) - c_0]$.

Barend J. Thijsse *is Antoni van Leeuwenhoek professor in the Department of Materials Science and Engineering at Delft University of Technology. His current interests focus on computational materials science and include electronic and atomic-scale modeling, studies of solid-state transformations, nanoscale mechanics, dynamics of thin-film behavior, and alloy design. Thijsse has a PhD in experimental molecular physics from the University of Leiden. Contact him at b.j.thijsse@tudelft.nl.*

Bert W. Rust *is a mathematician at the US National Institute for Standards and Technology. His research interest include ill-posed problems, time-series modeling, nonlinear regression, and observational cosmology. Rust has a PhD in astronomy from the University of Illinois. He is a member of SIAM and the American Astronomical Society. Contact him at bwr@nist.gov.*