# Comment

**Leo Breiman**

It is a pleasure to comment on papers where one finds much of merit together with major issues on which one can contend. This is the case with Ramsay's work. I have become impressed with the usefulness of splines over the past few years. They have nice theoretical properties as well as being a good practical tool in data analysis. Their use in statistics is not widespread, so I welcome this article because it may lead to increased interest and applications.

Now, as to my comments:

Ramsay's example on ACE will probably strike terror into the hearts of hundreds of contented ACE users. His ACE runs on the gasoline consumption data purport to show an extreme dependence on the order in which variables are entered into ACE.

We have used the same data (kindly published in the article) and are completely unable to replicate his results. We thought he might be using the monotone option in ACE, so we ran it this way. Then we ran it without the monotone option. We also ran ACE on the data with a 40% fixed window size, which (see below) is roughly equivalent to using one interior knot.

In all three cases we changed the order in which variables were entered and compared the results. These are given in Figure 1. The differences, due to the change in order of entering variables, is miniscule. I don't understand how Ramsay got the results he did, but my best guess is that some mistake was made in doing his ACE runs.

The ACE algorithm has been circulated and used far and wide over the last few years. In practice, it has proven a generally robust and illuminating data analysis tool.

I am very wary of the assumption of monotonicity. I was, for instance, against the inclusion of monotizing transformations as an option in the ACE algorithm. But my co-author, Jerome Friedman, argued me into it asserting that, whatever the reason, lots of statisticians like monotone transformations.

No one has decreed that phenomena in nature are inherently monotone. By restricting yourself to monotone transformations, you risk missing some important discoveries. For example, in the ACE paper (Breiman and Friedman, 1985) we give an example of ACE runs on an air pollution data set where one of the predictor variables is the pressure difference at two meteorological stations. One is in the Los Angeles basin and the other about 30 miles to the north.

In previous analyses of this data, many monotone transformations had been tried on this variable, and it always wound up as having very little predictive power. ACE produced a transformation on this variable that resembled $-|x|$. In this transformed mode, it became a strongly predictive variable. In retrospect, the reason is obvious. Any kind of pressure difference, either positive or negative, encourages a moving air mass, and reduces pollution in the Basin.

I only know of infrequent cases in which I would insist on monotone transformations. Finding nonmonotonicity can lead to interesting scientific discoveries. If the appropriate transformation is monotone, then the fitted spline functions (or ACE transformations) will produce close to a monotonic transformation. So it is hard to see what there is to gain in the imposition of monotonicity.

Ramsay claims that in practical applications a very small number of interior knots are sufficient. He cites two interior knots as being usually good enough. This is contrary to my experience.

The number of knots in a spline fit can be viewed as equivalent to the window size in a smoother. In fact, because spline fitting is a linear operation, one can compute the shape of the windows produced by spline fitting.

For one interior knot in a quadratic spline the windows are very broad, being equivalent (under a sensible definition) to a fixed window size smoother that uses about 35% of the data. For two interior knots, the equivalent window size is 29% of the data. For six interior knots the equivalent window size is 15%, and 10 interior knots drop it to 10% of the data.

Now a smoother with a 30% window will capture only the broadest gross features of the data. It will oversmooth the peaks and valleys and wipe out any salient fine features of the data. Unless one is willing to live with these oversmoothing characteristics, two knots are not sufficient.

The setting of an appropriate window size for the data is a critical and complex problem. In ACE it is done by using a smoother that selects a locally optimal window size ranging from 7% of the data to 30%. In smoothing splines it is done by choosing the optimal size of the penalty parameter through cross-validation. A large literature has developed in the theory and practice of choosing a good kernel sharpness

*Leo Breiman is Professor, Department of Statistics, University of California at Berkeley, Berkeley, California 94720.*

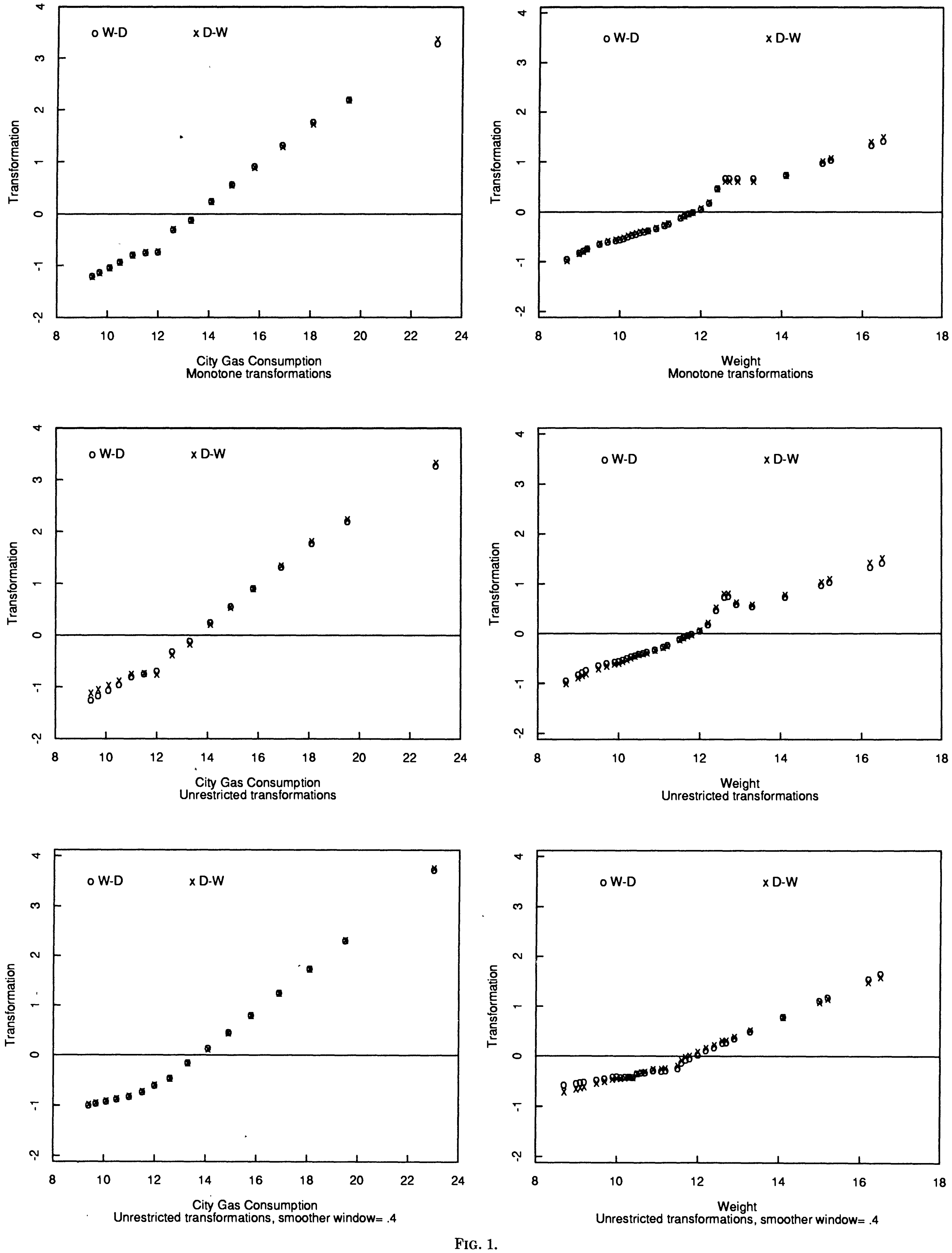
o W-D
x D-W
Transformation
-2
-1
0
1
2
3
4
8
10
12
14
16
18
20
22
24
City Gas Consumption
Monotone transformations
Weight
Monotone transformations
City Gas Consumption
Unrestricted transformations
Weight
Unrestricted transformations
City Gas Consumption
Unrestricted transformations, smoother window= .4
Weight
Unrestricted transformations, smoother window= .4

FIG. 1.

parameter in density estimation, which is simply another reflection of the window size problem.

Two things can be done in choosing the appropriate number of knots. One can assume the same number of knots on each variable, and, in predictive situations such as regression, use cross-validation to determine the optimal number. Burman (1988) has shown that this procedure has good asymptotic properties.

The procedure I favor and have extensively tested was suggested by Smith (1978). de Boor (1978) states that a wide variety of functions can be fitted with splines provided the knots are "well-chosen." My impression, after much experimentation, is the same—that few knots suffice providing that they are in the right place.

Unfortunately, direct numerical methods for moving knots around and trying to find optimal placements have not been very successful. Smith's method approaches knot placement in an indirect way that works well.

For bivariate data, the idea is this: place plenty of knot points along the $x$-axis. My experience is that the best placement is to have about equal number of data points between knots with knots at the minimum and maximum values of $x$.

Now regress $y$ against the truncated power function basis for the spline functions (I have been using cubic splines). The advantage in using this basis is that each coefficient corresponds to a single knot.

The next is to do classical deletion of variables. In this case, deleting a variable corresponds to deleting a knot. The deletion can either be done by best subsets or by simple backward deletion. In my version, I do simple backward deletion to keep the algorithm fairly efficient.

All knots are eventually deleted, Mallows' $C_p$ criterion is computed along the way, and the number of knots retained is determined by the minimum number of knots for which $C_p$ does not exceed its minimum value plus a small threshold. For 25 data points I use seven interior knots, and ten for 75 data points.

One interesting feature of this process is that it corresponds to a locally adaptive window size. Deleting a knot at a point has the effect of broadening the window size in the vicinity of that point. Thus what this process does is to keep small window sizes in the vicinity of rapid changes of the function and widen the window size elsewhere.

One other point might be interesting in terms of Ramsay's comments on the confidence intervals for the spline functions. The variance of most smoothing methods tends to funnel out near the end points of the data. This is a very natural result of the fact that near the end points the smooth around a point has to depend mainly on points to one side of it.

To cut down on this variability, I impose the condition that the spline function be linear at the end points, and as knots are removed adjacent to the end points that the linearity extend over the interval. There are two advantageous results. The first is that the funneling of variance effect at the end points is decreased. The second is that the spline basis consists only of the truncated power functions plus a linear and constant term.

Steve Peters and I have run a large simulation comparing four smoothers that do automatic window size selection (the above procedure, smoothing splines, supersmoother and a cross-validated kernel smoother). The modified Smith procedure did very well. It was able to track even relatively complex functions such as a function constant except for a sharp spike near the middle of the range.

This idea can be extended to multivariate procedures, and we are working on a spline version of ACE incorporating this approach. It is true that using the truncated power basis can lead to computational difficulties involving ill-conditioned matrices, but these are relatively easy to get around.

There is one disturbing aspect of splines I have come across. When using a fixed window size smoother on bivariate data, the window size, by very definition, stays constant as one moves across the data, except near the end points where it decreases. In this sense every interior point is given equal weight in determining the smooth.

But in fitting splines, the weights assigned to various points depend on their position relative to the knots. This can be most easily seen by graphing the equivalent window sizes versus $x(n)$, for equispaced $\{x(n)\}$. For two interior knots this graph has four maxima separating three minima. The heights of the two largest peaks are 35. The heights in the two lowest minima are 20. Thus, in the spline fit some points are weighted almost double that of others.

I don't know what effect this has on the fit. But a simple fix to this effect would be desirable. More research on this potential problem would be helpful.

The class of monotone cubic splines is not the integral of the class of $M$-splines with non-negative coefficients. A quadratic spline can be non-negative and still have negative coefficients in the $M$-spline basis (a look at Figure 1 in the article is convincing). For linear splines, simple linear inequality constaints on the coefficients can insure non-negativity. For quadratic splines, the constaints become nonlinear in the coefficients.

It is not clear what the effect is of working only within this restricted class of monotone splines, or what monotone splines are ruled out. The author's procedure gives simpler and more efficient optimization algorithms, as compared to allowing all monotone splines and optimizing under the nonlinear constraints. But I remain uncomfortable about the exclusion.

In describing the fitting of the yarn data, the author states that "The fitting criterion could be least squares, but this is not desirable when the dependent variable is being transformed," and he opts for maximum likelihood or Bayesian approaches; in this instance for maximum likelihood.

The reason why least squares is not desirable is not stated. Least squares is an old and reliable friend. Maximum likelihood or Bayesian approaches always impose distributional assumptions on the data which are usually difficult to verify. If you clap when Tinkerbell asks "do you believe in fairies" then fine. If you are in doubt, as I often am with real data, then use an earthy friend.

Ramsay states "Interval estimates for the transformation and structural parameters can be obtained by asymptotic techniques, jackknifing or bootstrapping." After this is stated, he passes by. But this is a very moot point. The ability of any of these to give accurate confidence intervals for finite data sets in the present context has never, to my knowledge, been thoroughly investigated.

In particular, the ability of jackknifing and of bootstrapping, or of any resampling method, to give accurate answers in complex regression situations is still in doubt.

In the same section, Ramsay refers to the problem of confidence intervals when some parameters are zero. In his regression set-up, if the transformation on the response is held constant, then the problem is that of least squares regression where a non-negativity constraint is put on the coefficients. In the solution some parameters are zero and the rest are positive.

Ramsay's suggestion is to treat the zero coefficients as being fixed at zero (i.e., out of the model) and to derive confidence intervals for the remainder. One difficulty with this is that if a new data set is sampled from the same underlying distribution and the procedure repeated, then a different set of parameters may be zeroed, and parameters formerly zero may have positive values.

A very similar question is involved in deletion of variables in a regression. The usual procedure for deriving confidence intervals (in statistical packages) is to do the deletion. Then after the number of variables to be retained is somehow decided, proceed as if these were the only variables around in the first place and derive standard confidence intervals based on those variables remaining in the equation.

This is a highly questionable and certainly biased procedure. If for nothing else, what are the confidence intervals for the coefficients of the variables deleted in forming the final reduced equation? I don't know of any satisfying answers.

## ADDITIONAL REFERENCES

BURMAN, P. (1988). Rates of convergence of the estimates of optimal transformations and correlations in regression. Unpublished.

SMITH, P. (1978). Curve fitting and modelling with splines using statistical variable selection techniques. NASA Contractor's Report 166034.

## ADDED NOTE

Mainly due to Professor Ramsay's efforts, we have resolved the issue of the different ACE outputs. A few months after we began releasing copies of ACE, my co-author, Jerome Friedman, added a subroutine called SCALE to the code.

SCALE does a linear regression of $y$ on the independent variables $x_1, \cdots, x_m$, getting the regression equation $\sum \beta_m x_m$. Then ACE takes the initial value of the transform of $x_m$ to be $\beta_m x_m$ (all means have been subtracted). In the original code, the initial transform of $x_m$ is identically zero. I was running the later version, and Professor Ramsay had the earlier version. When he ran the code I sent him, his results were similar to mine. In this example, with almost linear transforms and heavy correlation between the two predictor variables, it is clear that the use of SCALE would stabilize the ACE results.