SMOOTHING LOCALLY REGULAR PROCESSES BY BAYESIAN NONPARAMETRIC METHODS, WITH APPLICATIONS TO ACID RAIN DATA ANALYSIS

by

HON WAI MA
B.Sc., CONCORDIA UNIVERSITY, 1984

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE

in

THE FACULTY OF GRADUATE STUDIES
Department of Statistics

We accept this thesis as confirming
to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA
September 1986

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Statistics

The University of British Columbia
1956 Main Mall
Vancouver, Canada
V6T 1Y3

Date Sept 18, 1986

ABSTRACT

We consider the problem of recovering an unknown smooth function from the data using the Bayesian nonparametric approach proposed by Weerahandi and Zidek (1985). Selected nonparametric smoothing methods are reviewed and compared with this new method. At each value of the independent variable, the smooth function is assumed to be expandable in a Taylor series to the pth order. Two methods, cross-validation and "backfitting" are used to estimate the a priori unspecified hyperparameters. Moreover, a data-based procedure is introduced to select the appropriate order p. Finally, an analysis of an acid-rain, wet-deposition time series is included to indicate the efficacy of the proposed methods.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to thank Dr. Harry Joe for his guidance and assistance in the producing of this thesis. I am indebted to Dr. James Zidek for his helpful suggestions, encouragement, and for his careful reading of this thesis. I also appreciate the assistance of Brian Leroux in translating the article of Collomb (1981).

The finanical support of the Department of Statistics of the University of British Columbia is gratefully acknowledged.

1. INTRODUCTION.

This thesis is concerned with a Bayesian nonparametric method for estimating an unknown smooth function given some observed data with noise.

Consider the model $Y(x_i) = S(x_i) + \epsilon_i, \quad i = 1, ..., n$ where $S(\cdot)$ is an unknown smooth function and the random variables ϵ_i are uncorrelated with zero mean and constant variance σ^2. In recent years, many smoothing methods have been developed to estimate S nonparametrically from the observations on $Y(x_i), \quad i = 1, ..., n$.

A survey of selected nonparametric smoothing methods is presented in section 2. Included are the kernel method, the spline smoothing method, the exponential smoothing method and some robust smoothing methods.

Weerahandi and Zidek (1985) propose a Bayesian nonparametric method to tackle such problems. The method is flexible and easy to use. Its derivation is technically elementary and can be extended to the analysis of space-time series and the regression problem. S is assumed to be locally regular, that is expandable in a Taylor series about $x = x_{n+1}$ to the pth term. The degree of local regularity is reflected in the size of p. The objects of inference are an estimate of $S(x)$ and a corresponding credibility interval, where x may lie outside the range of the data. The general theory is described in section 3 and a brief comparison of their approach to those mentioned in section 2 is also included.

This thesis is a supplement to the work of Weerahandi and Zidek (ibid). Besides using the cross-validation method to estimate the a priori unspecified hy-

perparameters, a new "backfitting" method is proposed as an alternative. The asymptotic behaviour of the cross-validated sum of squared errors is also examined. A data-based procedure is developed to choose the appropriate order p. Section 3 gives detailed discussions of these proposed methods.

Finally, in section 4, all the results are tested on the data of an acid-rain wet-deposition time series. The calculations were performed by using Fortran subroutines which are included in the Appendix.

2. SURVEY OF SELECTED NONPARAMETRIC SMOOTHING METHODS.

Summary : Suppose we are given observations $\left[\left(x_i, Y(x_i)\right), \quad i = 1, ..., n\right]$ satisfying the model

$$Y(x_i) = s(x_i) + \epsilon_i, \qquad i = 1, ..., n. \tag{2.1.1}$$

where the $\{\epsilon_i\}$ are independent and identically distributed (i.i.d.) random variables with zero mean and constant variance σ^2. The abscissae $\{x_i\}$ are assumed to be known without error, and to satisfy $x_1 \leq x_2 \leq ... \leq x_n$. Here $s(\cdot)$ is a fixed but unknown smooth function.

We consider the problem of recovering $s(\cdot)$ when only discrete, noisy measurements of it are available. In recent years, many nonparametric smoothing methods have been developed to tackle this problem. The extensive literature in this area makes a detailed review of each paper infeasible. We aim instead for a brief summary of the principal nonparametric smoothing methods including the kernel method, the spline smoothing method and the exponential smoothing method which is used quite often in the analysis of time-series data. Finally, some robust smoothing methods are also considered. This survey will help to put the developments of section 3 into perspective.

2.1 The Kernel (Moving Window) Method

Watson (1964) and Nadaraya (1964) independently and simultaneously proposed the kernel estimator of $s(\cdot)$:

$$\hat{s}(x) = \frac{\sum_{i=1}^{n} Y(x_i) K\left((x - x_i)/h\right)}{\sum_{i=1}^{n} K\left((x - x_i)/h\right)}$$

where the kernal $K(x)$ is a certain density function satisfying the conditions :

$$\sup_{-\infty<x<\infty} |K(x)| < \infty, \quad \lim_{x\to\pm\infty} |xK(x)| = 0, \quad \int_{-\infty}^{\infty} |K(x)|\,dx < \infty,$$

and $h = h(n)$ (the bandwidth of the kernel), a sequence of positive numbers converging to zero which specifies the degree of smoothness of the estimator.

This is analogous to the kernel density estimator introduced by Rosenblatt (1956). He proposed the kernel estimator, which can be written as

$$\hat{f}_n(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

where K is the kernel and $x_1, ..., x_n$ are assumed to be i.i.d. with the unknown density $f(\cdot)$.

Referring back to the Watson-Nadaraya estimator, Watson (ibid), using some Monte Carlo experiments, concluded that it is h and not K which is important in the estimate. Schuster (1972) obtained asymptotic normality as a correction of a result of the same type given by Nadaraya. A related estimator was introduced by Priestly and Chao (1972) :

$$\hat{s}(x) = \frac{1}{h}\sum_{i=1}^{n} Y(x_i)(x_i - x_{i-1})K\left(\frac{x - x_i}{h}\right)$$

where K is the kernel, satisfying the conditions :

$$K(u) \geq 0, \text{ for all } u, \quad \int_{-\infty}^{\infty} K^2(u)\,du < \infty, \text{ and } K(u) = K(-u).$$

Stone (1977) considered a general form of estimator,

$$\hat{s}(x) = \sum_{i=1}^{n} w_{ni}(x)Y(x_i),$$

where $w_{ni}(x) = w_{ni}(x, x_1, ..., x_n)$, $1 \leq i \leq n$, depends more heavily on those x_i close to x. This formulation includes the Watson-Nadaraya and the Priestly-Chao estimators as special cases. Stone (ibid) also derived the sufficient conditions on $\{w_n\}$ for the consistency of $\hat{s}(x)$.

Clark (1977) proposed a convolution-based estimator,

$$\hat{s}(x) = \int_{-\infty}^{\infty} g(t) \frac{1}{h} K\left(\frac{x-t}{h}\right) dt,$$

where

$$g(x) = \begin{cases} Y_1 & \text{for } x \leq x_1, \\ Y(x_i)\left(\frac{x_{i+1}-x}{x_{i+1}-x_i}\right) + Y(x_{i+1})\left(\frac{x-x_i}{x_{i+1}-x_i}\right) & \text{for } x_i \leq x \leq x_{i+1}, \\ Y_n & \text{for } x \geq x_n. \end{cases}$$

He claimed that such estimators have some advantages over the Watson-Nadaraya and the Priestly-Chao estimators in small samples. Firstly, $\hat{s}(x)$ is smoother. Secondly, the Watson-Nadaraya and the Priestly-Chao estimators are not necessary invariant under linear transformations and they do not have the shape-preserving properties of $\hat{s}(x)$.

It is well known [see Table 1 of Rosenblatt (1971)] that the choice of the kernel function K, is of relatively unimportance. Usually, a kernel is chosen for computational convenience. While the choice of kernel does not seem to be an issue, at least for reasonably moderate sample sizes, the choice of bandwidth is quite a different matter. In practice, the estimate is often formed by computing the estimate for several h's and selecting the estimate with the most pleasing appearance. If the h is too small, the estimate will appear to be too wiggly and if too large, the estimate will begin to be too flat.

Clark (1977) estimated h from the data by a minor modification of the cross-validation technique described by Wahba and Wold (1975). A simulation experiment demonstrated that this approach was satisfactory in determining the appropriate degree of smoothness. Similarly, Hardle and Marron (1985) estimated h of the Watson-Nadaraya estimator by minimizing

$$CV(h) = n^{-1} \sum_{i=1}^{n} \big(Y(x_i) - \hat{s}_i(x_i)\big)^2$$

where $\hat{s}_i$ is the cross-validated estimator given by

$$\hat{s}_i(x) = \frac{\sum_{i \neq j} Y(x_j) K((x - x_j)/h)}{\sum_{i \neq j} K((x - x_j)/h)}.$$

They also showed that such an h is asymptotically optimal with respect to the distances :

$$MSE \text{ (Averaged Squared Error)} = n^{-1} \sum_{i=1}^{n} \big(\hat{s}(x_i) - s(x_i)\big)^2,$$

$$ISE \text{ (Integrated Squared Error)} = \int \big(\hat{s}(x) - s(x)\big)^2 f(x)\, dx$$

where $f(x)$ is the marginal density of x, and

$$CMISE \text{ (Conditional Mean Integrated Squared Error)} = E[\ ISE \mid x_1, ..., x_n\ ].$$

We may be interested in constructing confidence intervals for $s(x)$ or confidence bands for $s(\cdot)$. However, besides some theoretical results mentioned in Bickel and Rosenblatt (1973) and Collomb (1981), no practical method has been reported.

2.2 The Spline Smoothing Method

For the model in equation (2.1.1), the spline smoothing estimate s_α of s is defined to be the curve that minimizes

$$\frac{1}{n}\sum_{i=1}^{n}\big(Y(x_i)-s(x_i)\big)^2+\alpha\int s''(u)^2\,du \qquad (2.2.1)$$

over the class of all functions s for which s and s' are absolutely continuous and s'' is square-integrable. The idea of (2.2.1) is to trade off fidelity to the data (small sum of the squared residuals) against smoothness (low value of integrated square second derivative). The smoothing parameter α determines the rate of exchange between these quantities and hence determines the amount by which the data are smoothed to produce the estimate s_α. As $\alpha \to \infty$, s_α becomes linear, and the limiting function s_∞ is the least squares straight line passing through the data. As $\alpha \to 0$, $n^{-1}\sum_{i=1}^{n}\big(Y(x_i)-s_\alpha(x_i)\big)^2 \to 0$, if no x_i's are replicated, until, in the limit, s_0 passes through the data; s_0 is the natural cubic spline interpolator of the data which is discussed below. The idea underlying this estimate can be traced back to Whittaker (1923), who proposed a discrete version using finite differences in place of derivatives in (2.2.1). Schoenberg (1964) proposed the replacement of the finite differences in Whittaker's objective function by derivatives.

Reinsch (1967) showed that s_α is a cubic spline with the following properties : (i) it is a cubic polynomial in each interval (x_i, x_{i+1}), (ii) at each x_i, the curve and its first two derivatives are continuous, but there may be a discontinuity in the third derivative, (iii) in each of the ranges $(-\infty, x_1)$ and (x_n, ∞) the second derivative is zero, so that s_α is linear outside the range of the data.

The method of cross-validation (see, for example, Stone (1974)) has been suggested for choosing the smoothing parameter α from the data. Let $s_\alpha^{-i}(x_i)$ be

the solution to problem (2.2.1) with the ith data point , $(x_i, Y(x_i))$, omitted. Of course, $s_\alpha^{-i}(x_i)$ may be regarded as an estimator of $Y(x_i)$ and α as appropriately chosen if $s_\alpha^{-i}(x_i)$ is a good estimator of $Y(x_i)$. To measure the goodness of fit, we choose the average squared error, that is ,

$$XVSC(\alpha) \stackrel{\text{def}}{=} n^{-1} \sum_{i-1}^{n} \left(Y(x_i) - s_\alpha^{-i}(x_i)\right)^2 \tag{2.2.2}$$

which is called the cross-validation score. Let $A(\alpha)$ be a matrix such that

$$s_\alpha(x_i) = \sum_j A_{ij}(\alpha) Y(x_j).$$

Craven and Wahba (1979) showed that (2.2.2) has the easier computational form

$$XVSC(\alpha) = n^{-1} \sum_{i=1}^{n} \frac{\left(Y(x_i) - s_\alpha(x_i)\right)^2}{\left(1 - A_{ii}(\alpha)\right)^2}. \tag{2.2.3}$$

They also suggested the use of a related criterion, called generalized cross-validation, obtain from (2.2.3) by replacing $A_{ii}(\alpha)$ by its average value, $n^{-1}\text{trace}A(\alpha)$, say, $n^{-1}\text{tr}A(\alpha)$. This gives the score

$$GXVSC(\alpha) = n^{-1} RSS(\alpha) / \left(1 - n^{-1}\text{tr}A(\alpha)\right)^2, \tag{2.2.4}$$

where $RRS(\alpha)$ is the residual sum of squares $\sum_{i=1}^{n} \left(Y(x_i) - s_\alpha(x_i)\right)^2$. Craven and Wahba gave fuller details and justification of the preceding derivation. Their arguments suggest that generalized cross-validation should yield approximately the optimal value for α, in the sense of minimizing the average squared errors $n^{-1} \sum_{i=1}^{n} \left(s_\alpha(x_i) - s(x_i)\right)^2$. (A Fortran implementation can be found in the IMSL Library (1980).)

To use the formula (2.2.4) one still needs to find the $\text{tr}A(\alpha)$. Silverman (1984) proposed a way to derive such an approximation. Let $\hat{f}(x)$ be an estimate of the local density of x_i, calculated using the fast algorithm of Silverman

(1982) on the range $[a,b]$ just containing the design points. Let

$$c_0 = \pi^4 n^{-1} \Big(\int_a^b \hat{f}(x)^{1/4}\, dx \Big)^{-4}.$$

It can then be shown that

$$\mathrm{tr}A(\alpha) \approx 2 + \sum_{i=3}^{n} \big(1 + c_0 \alpha (i - 1.5)^4\big)^{-1}. \tag{2.2.5}$$

Substituting (2.2.5) into (2.2.4) gives a score function called the asymptotic generalized cross-validation ($AGXV$) score which can then be minimized to give an automatic choice of smoothing parameter. A simulation study is carried out in Silverman (1984) and the practical performance of $AGXV$ turns out to be even better then that of generalized cross-validation in that, for non-uniformly spaced data, the proportion of cases giving a bad value of the smoothing parameter is substantially reduced.

In constructing confidence intervals, Wahba (1983) exploited a Bayesian interpretation of the smoothing spline, that is a Bayes estimate with respect to a certain zero mean Gaussian prior. She was able to develop credibility intervals for the $s(x_i)$. However, it should be emphasized that these are not simultaneous credibility interval estimates. Some extensions and modifications of Wahba's work can be found in Silverman (1985).

2.3 The Exponential Smoothing Method

Exponential smoothing method was developed in the early 1950s. Since then it has become a popular method of forecasting. It is because the model formulations are relatively simple. Also, only limited data storage and computational effort are required. Perhaps the most important reason for the popularity

of exponential smoothing is the surprising accuracy that can be obtained with minimal effort in model identification. Makridakis et al. (1982) reported that exponential smoothing method proved to be at least as accurate for forecasting as more complex approaches such as the Box-Jenkins autoregressive integrated moving average (ARIMA) technique.

In essence exponential smoothing method applies an unequal set of weights to past data. These weights decay in an exponential manner from the most recent data value to the most distant value. It is also assumed that the $\{x_i\}$ are equally spaced and that $Y(x_i) = Y_i = Y_t$ is a time series. The main use is to forecast Y_{t+1} given $Y_1, ..., Y_t$ although this method can also be used to smooth $Y_1, ..., Y_n$ (for example, see Fig. 1 in Brown and Meyer (1961)). However, it should be pointed out that this method depends just on the previous data. So as a smoothing method, it is probably not as good as the kernel and the spline smoothing methods which use the data in a neighbourhood of a point. Here are some commonly used model formulations:

(i) **Simple exponential smoothing** (no trend in the series)

The general equation for the forecast value can be written as

$$F_{t+1} = \alpha Y_t + (1 - \alpha) F_t \tag{2.3.1}$$

where F_t is the forecast at time t, Y_t is the observed value of the series at time t and α is a smoothing constant between 0 and 1. A large value of α, say .9, gives little smoothing in the forecast, where a small value of α, say .1, gives considerable smoothing. Most often α is chosen by trial-and-error. An optimal value of α may be chosen by minimizing the MSE of the forecast. The

initial value (as of time 0), F_0, can be estimated simply using the mean of the data. Alternatively, backcasting is suggested to obtain F_0. It is done by inverting the data series and starting the estimation procedure from the latest (most recent) value and finishing with the first (oldest) value. Doing this will provide forecasts or parameter estimates for the beginning of the data, and these can be used as initial values when the data are forecast in the usual sequence, that is, from the beginning to the end.

Trigg and Leach (1967) suggest an adaptive response rate approach for the automatic specification of α. It allows α to vary from one period to another so that the forecast can respond to the pattern of the data as closely as possible. The adaptive response rate approach is based on (2.3.1), but α varies and is calculated according to

$$\alpha_{t+1} = |E_t/M_t|$$

where

E_t (smooth error) $= \beta e_t + (1-\beta)E_{t-1}$,

M_t (smooth absolute error) $= \beta|e_t| + (1-\beta)M_{t-1}$,

$e_t = Y_t - F_t$, and

β is a constant usually set at 0.1 or 0.2.

We can let the initial values $E_1 = 0$ and M_1 equal an estimate of the mean absolute deviation, that is,

$$MAD \stackrel{\text{def}}{=} \frac{1}{n}\sum_{i}^{n}|e_i|.$$

The value of α_t will fluctuate between 0 (perfect forecasts) and 1 (extreme bias in the forecasts).

(ii) **Winters' (1960) linear exponential smoothing** (linear trend in the series)

A smoothed estimate of the trend is given by

$$T_t = \beta(S_t - S_{t-1}) + (1-\beta)T_{t-1}, \tag{2.3.2}$$

where S_t is the equivalent of the simple exponential smoothed value and β is a smoothing constant, analogous to α. The initial values S_0 and T_0 can be obtained by backcasting. Alternatively, they are found by using ordinary least squares. It is done by solving the equation for a straight line to obtain the intercept and the slope, and using these as initial parameter values.

The basic principle underlying (2.3.2) is the same as that of (2.3.1). The most recent trend, $(S_t - S_{t-1})$, is weighted by β and the last smoothed trend, T_{t-1}, is weighted by $(1-\beta)$. Then this smoothed trend is combined with the standard smoothing equation to obtain

$$S_t = \alpha Y_t + (1-\alpha)(S_{t-1} + T_{t-1}). \tag{2.3.3}$$

In order to use these smoothed series values, S, and the smoothed trend component, T, to prepare a forecast, the trend component must be added to the basic smoothed value, for the number of periods m ahead to be forecast. The forecast is given by

$$F_{t+m} = S_t + mT_t.$$

(iii) **Winters' (1960) linear and seasonal exponential smoothing** (linear trend and seasonality in the series)

This model is based on three equations, each of which smooths a factor

associated with one of the three components of the pattern – randomness, linear, and seasonal. These three equations are as follows :

$$S_t = \alpha Y_t / I_{t-L} + (1-\alpha)(S_{t-1} + T_{t-1}),$$
$$T_t = \beta(S_t - S_{t-1}) + (1-\beta)T_{t-1}, \text{ and}$$
$$I_t = \gamma Y_t / S_t + (1-\gamma) I_{t-L},$$

where S is a smoothed value of the deseasonalized series, T is a smoothed value of the trend, I is a smoothed value of the seasonal factor, and L is the length of seasonality (e.g. number of months or quarters in a year). The initial value of the smoothed seasonal factor I can be found by the classical time series decomposition methods (for example, see Makridakis and Wheelwright (1978).)

The equation for I is comparable to a seasonality index. It must be remembered that Y_t includes any randomness in the series. In order to smooth this randomness, the equation for I weights the newly computed seasonal factor (Y_t/S_t) with γ and the most recent seasonal number corresponding to the same season, I_{t-L}, with $(1-\gamma)$. The equation for T_t is given in (2.3.2). In the equation for S_t, Y_t is divided by the seasonal factor I_{t-L}. This is done to deseasonalize (eliminate the seasonal fluctuations of) Y_t. I_{t-L} is used because I_t cannot be calculated until S_t is known. The forecast is given by

$$F_{t+m} = (S_t + mT_t) I_{t-L+m}.$$

For models (ii) and (iii), the best values of α, β and γ are found by trying various combinations of values and choosing that set of values that minimizes MSE of the forecast.

Only exponential smoothing models with simple patterns are described in the

above discussion. Smoothing models for other complicated patterns are desribed by Makridakis and Wheelwright (1978), Makridakis et al. (1982) and Gardner (1985).

2.4 Some robust smoothing methods

For the model in (2.1.1), Cleveland (1979) proposed the use of a robust weighted regression procedure for smoothing scatterplots. Firstly, for each i compute the estimates, $\hat{\beta}_j(x_i)$, $j = 0, 1, ..., d$ of the parameters in a polynomial regression of degree d of $Y(x_k)$ on x_k, which minimize

$$\sum_{k=1}^{n} w_k(x_i)\big(Y(x_k) - \beta_0 - \beta_1 x_k - ... - \beta_d {x_k}^d\big)^2$$

where $w_k(x_i) = W({h_i}^{-1}(x_k - x_i))$ and h_i is the rth smallest number among $|x_i - x_j|$, for j = 1,...,n. W is some given weight function and $r = fn$ where f is the parameter used to determine the amount of smoothing. Thus, $\hat{Y}(x_i)$, the fitted value of the regression at x_i, is $\sum_{j=0}^{d} \hat{\beta}_j(x_i){x_i}^j$.

Let δ_k be some robustness weights. Then compute a new $\hat{Y}(x_i)$ for each i by fitting a dth degree polynomial using weighted least squares with weight $\delta_k w_k(x_i)$ at $\big(x_k, Y(x_k)\big)$. Now repeat this procedure successively to compute new δ_k (by the procedure indicated below, for example) and $\hat{Y}(x_i)$, t times; the final $\hat{Y}(x_i)$ are robust locally weighted regression fitted values. Lastly, linear interpolation of these fitted values are used to obtain the robust smooth curve.

Cleveland chose $\delta_k = B(e_k/6s)$, where B, the bisquare weight function is defined by

$$B(x) = \begin{cases} (1-x^2)^2 & \text{for } |x| < 1, \\ 0 & \text{for } |x| \geq 1, \end{cases}$$

$e_k = Y(x_k) - \hat{Y}(x_k)$, and s is the median of the $|e_i|$. He also gave some guidelines for choosing f, d, t and W in practice. (Cleveland's procedure is available in S [Becker and Chambers (1984)] through the command "lowess".)

Friedman (1984) constructed a variable span smoother based on locally linear fits. It is very fast to compute and the value of the parameter that controls the amount of smoothing is automatically optimized locally (through cross-validation) , allowing it to adapt to the response over the range of abscissa values.

In the context of robust spline smoothing, we can replace the sum of squared errors in the objective function (2.2.1) by a different function of the errors, to give

$$\sum_{i=1}^{n} \rho\big(Y(x_i) - s(x_i)\big) + \alpha \int s''(u)^2 \, du. \tag{2.4.1}$$

Here ρ is a convex and less rapidly increasing function with bounded derivative.

Huber (1979) chose

$$\rho(x) = \begin{cases} \frac{1}{2}x^2 & \text{for } |x| \leq c, \\ c|x| - \frac{1}{2}c^2 & \text{for } |x| > c. \end{cases}$$

The constant c regulates the degree of robustness and good choices for c are between 1 or 2 times the standard deviation of the individual observation $Y(x_i)$. Lenth (1977) and Huber (1979) use iterative reweighted least squares procedures to mininize (2.4.1). Utreras (1981) gives a numerical procedure based on the Newton minimization algorithm for computing the minimizer of (2.4.1).

2.5 The Multivariate Extension

What we have discussed so far are the nonparametric smoothing methods of an univariate function $s(\cdot)$. If $s(\cdot)$ is now a function of a vector variable x, then the extension of the above smoothing methods is straightforward. Here is a brief description:

(i) the kernel method

Using the Watson-Nadaraya estimator as an example, now $h = (h_1, ..., h_p)$ $\in R_+^p$, the collection of p-tuples of positive numbers; $x = (t_1, ..., t_p) \in R^p$ and the kernel K is a certain density function on R^p. We can use the approach of Hardle and Marron (1985) described in section 2.1 to estimate h.

Bierens (1983) proposed a new class of kernel estimator which is constructed through a Sample Moments Integrating Normal Kernel (SMINK) estimator. The SMINK estimator is itself a density and satisfies the condition that it's first and second moment integrals equal the first and second sample moments, respectively. For $p = 2$, the kernel estimator of $s(\cdot)$ is given by

$$\hat{s}(x|h_1, h_2) = \frac{\int_{-\infty}^{+\infty} Y \hat{f}_1(Y, x|h_1)\, dy}{\hat{f}_2(x|h_2)}$$

where $\hat{f}_1(Y, x|h_1)$ and $\hat{f}_2(x|h_2)$ are SMINK estimators of the density $f(Y, x)$ and $f_2(x)$, respectively, and h_1 and h_2 are kernel bandwidth parameters. A two-step procedure is used to estimate h_1 and h_2 and satisfactory performance of $\hat{s}(x)$ is revealed in a numerial example.

(ii) the spline smoothing method

For the model in (2.1.1), s is now a function of a p-dimensional vector variable x, where $x = (t_1, ..., t_p)$. The smoothness penalty function $\int s''(u)^2\, du$ in (2.2.1) is replaced by

$$J_m(s) = \sum_{\gamma_1+...+\gamma_p=m} \frac{m!}{\gamma_1!, ..., \gamma_p!} \int ... \int \left(\frac{\partial^m s}{\partial t_1^{\gamma_1} ... \partial t_p^{\gamma_p}}\right)^2 dt_1 ... dt_p.$$

In the particular case, $p = m = 2$, the smoothness penalty function becomes

$$J_2(s) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(\frac{\partial^2 s}{\partial t_1^2} + \frac{2\partial^2 s}{\partial t_1 \partial t_2} + \frac{\partial^2 s}{\partial t_2^2}\right) dt_1\, dt_2.$$

We wish to find the multivariate spline smoothing estimate, s_α, of α which minimizes

$$\frac{1}{n}\sum_{i=1}^{n}\left(Y(x_i) - s(x_i)\right)^2 + \alpha J_m(s). \tag{2.5.1}$$

For $p = m = 2$, (2.5.1) becomes

$$\frac{1}{n}\sum_{i=1}^{n}\left(Y(x_i) - s(x_i)\right)^2 + \alpha J_2(s). \tag{2.5.2}$$

The minimizer of (2.5.2) is one of the "thin plate splines", so called because $J_2(s)$ is the bending energy of a thin plate.

Wahba and Wendelberger (1980) and Wahba (1980) have shown how to use generalized cross-validation (see Craven and Wahba (1979)) to choose α from noisy data, and have given a computational algorithm for $p = 2$. Utreras (1979) has also given computational procedures. In the general case, i.e., estimating the s_α which minimizes (2.5.1), Wahba and Wendelberger (1980) and Wahba (1984a,b) do provide solutions.

It may be noteworthy that in the mining industry, a class of two dimensional smoothing methods known as "kriging" are closely related to the thin-plate splines. Comparisons of these two smoothing methods are found in several sources. Interested readers are referred to Dubrule (1983,1984) and Watson (1984).

(iii) the exponential smoothing method

Simple exponential smoothing method has been generalized to multivariate forcasting by Jones (1966) and Enns et al. (1982). The generalization is straightforward. We replace the scalars in equation (2.3.1) with:

$$F_{t+1} = \alpha Y_t + (1 - \alpha) F_t.$$

If there are p series, the dimensions of F_t, F_{t+1} and Y_t are $p \times 1$. The smoothing matrix α has dimension $p \times p$.

Jones (ibid) proposes a recursive algorithm to estimate the smoothing matrix α. Enns et al. (ibid) introduce a class of multiple exponential smoothing models as an alternative to simple univariate exponential smoothing and Trigg and Leach (1967) suggest adaptive models. Such models are adaptive in that the smoothing matrix α, which is a generalization of the smoothing constant of univariate models, changes from period-to-period. The model parameters, including the full variance-covariance structure of the multiple time series as well as the smoothing matrix α are estimated by maximum likelihood.

Harvey (1984) further simplified the computations of Enns et al. (ibid) so that one can use univariate smoothing models to forecast related time series. Moreover, Harvey's results hold for smoothing models containing polynomial trends

and seasonal components as well.

(iv) the robust smoothing methods

To our knowledge, no generalization to the multivariate analysis has been reported up to now.

2.6 Discussion

It should be emphasized that in the above survey, we have been guided in our choice of material by concerns related to the applicability of the methodology rather than theoretical properties of different kinds of smoothers. For discussion related to the latter and comprehensive lists of references on this subject, please refer to Stone (1977), Wegman (1980), Collomb (1981), Wegman and Wright (1983) and Silverman (1985).

From this survey, we can discover some shortcomings of the described smoothing methods. For instance, the spline smoothing and the robust smoothing methods are used for smoothing rather than forecasting. The exponential smoothing method is mainly used for forecasting but not for smoothing. Also, explicit form of confidence intervals of the smooth curve estimate are hard to derive. However, Weerahandi and Zidek (1985) provide an alternative approach to solve these problems. The general theory is presented in details in the following section.

3. THEORY OF BAYESIAN NONPARAMETRIC APPROACH FOR SMOOTHING LOCALLY REGULAR PROCESSES.

3.1 The General Theory

Weerahandi and Zidek (1985) (abbreviated hereafter by WZ) propose a Bayesian nonparametric approach for the smoothing of stochastic processes. The stochastic processes of concern are of the form $R = S + N$ where S is a smooth function and N is an independent noise process. In this thesis we take the domain to be the real line and may think of the stochastic process as a time series. Assume R has been observed at a sequence of time-values, t_i, and let $r_i = R(t_i) \quad i = 1, ..., n$. The objects of inference are an estimate, $\beta_0 = S(t_{n+1})$, the interpolant or predictand, and a corresponding credibility interval. S is assumed to be locally regular, that is expandable in a Taylor series to the pth term about $t = t_{n+1}$. A local parametrization of S is thereby achieved where the parameters are $\beta_i = D^i S(t_{n+1})$, D denoting the differentiation operator. An a priori structural model for the data is

$$R = X\beta + \epsilon \qquad (3.1.1).$$

Here $R = (R_1, ..., R_n)^T$, $R_i = R(t_i)$, $\beta = (\beta_0, ..., \beta_p)^T$, X, an $n \times (p+1)$ matrix, is given by $X = (1, X_1, ..., X_p)$, 1 is an n-vector of 1's, ${X_j}^T = ([t_1 - t_{n+1}]^j/j! \ , ..., \ [t_n - t_{n+1}]^j/j!) \qquad j = 1, ..., p$, $\epsilon = \eta + N$, $N^T = (N_1, ..., N_n)$, $N_i = N(t_i)$, $\eta^T = (\eta_1, ..., \eta_n)$, and η_i is the remainder of the Taylor expansion of $S(t_i)$, that is $\eta_i = [t_i - t_{n+1}]^{p+1} D^{p+1} S(\theta_i)/(p+1)!$ where θ_i is a point in the interval joining t_i and t_{n+1}. The degree of local regularity is reflected in the size of p.

Although R is observed, β in equation (3.1.1) and ϵ are not. The sec-

ond assumption in WZ's approach is that the expansion errors and all other a priori uncertainity about R, β and ϵ have a joint multivariate normal distribution. Furthermore, the η_i's are assumed to be independent with mean zero and the variances of the η_i's are of the form of $\delta^2|t_i - t_{n+1}|^{2p+2}$ where δ^2 represents uncertainity about the size of $D^{p+1}S/(p+1)!$ in the remainder term of the Taylor expansion of S.

In order to estimate $\beta_0 = S(t_{n+1})$, the interpolant or predictand, assume that equation (3.1.1) holds with $\mathrm{Cov}(\epsilon) = \sigma^2 H$, where $\sigma^2 = \mathrm{Var}(N_i)$ is the variance of the noise, $H = \mathrm{diag}\{(1 + c|t_1 - t_{n+1}|^{2p+2}), ..., (1 + c|t_n - t_{n+1}|^{2p+2})\}$, $c = \delta^2/\sigma^2$, and $\mathrm{Cov}(\beta) = \Sigma$. Then the inference problem can be solved by using standard results (see Lindley and Smith (1972)). Let "$U|V$" denote the conditional distribution of U given V. Then

$$R|\Sigma, \sigma^2, H \sim N(X\beta^0, X\Sigma X^T + \sigma^2 H)$$

and

$$\beta|R, \Sigma, \sigma^2, H \sim N(F(\Sigma^{-1}\beta^0 + \sigma^{-2}X^T H^{-1} R), F)$$

where

$$F = (\Sigma^{-1} + \sigma^{-2}X^T H^{-1} X)^{-1}.$$

Then using a diffuse prior for β, i.e., letting $\Sigma^{-1} \to 0$, $E(\beta|R)$ converges to $(X^T H^{-1} X)^{-1} X^T H^{-1} R$, the generalized least squares estimator of β with a posteriori covariance matrix $\sigma^2(X^T H^{-1} X)^{-1}$.

It is noteworthy that c serves as the parameter controlling the appropriate degree of smoothing. If c is too small, $\hat{r}$ will appear to be too smooth, and if

too large, $\hat{r}$ will be too rough. One approach to estimate c is cross-validation (Stone (1974)). Successively, $t_{n+1} = t_i$ is chosen, and then r_i is dropped from the sample and interpolated using $\hat{r}_i = \hat{\beta}_0(i)$, obtained by the above generalized least squares procedure. Lastly, $c = \hat{c}$ is chosen to be the value which minimizes the cross-validated sum of squared errors,

$$XV(c) \stackrel{\text{def}}{=} \sum_i (r_i - \hat{r}_i)^2.$$

Alternatively, maximum likelihood estimation may be used to find c. However, in the acid rain example (see section 4), $\hat{c} = 0$ was obtained, so this method is ineffective and hence rejected. A "backfitting" method decribed below can also be used to estimate c. It is shown to be computationally efficient and it coheres to the general theory. A detailed description of this method is given in section 3.2 (II).

For $t_{n+1} = t_i$, the a priori local structural model is $R_i = \alpha_i + \sigma Z_i, \quad i = 1, ..., n$ where $\alpha_i = S(t_i)$ and Z_i has mean 0, variance 1. Given $c = \hat{c}$, $\hat{\alpha}_i$ and $\text{Var}(\hat{\alpha}_i) = c_i\sigma^2$ can be estimated by the above generalized least squares procedure. Thus $E(R_i - \hat{\alpha}_i)^2 = \text{Var}(\hat{\alpha}_i) + \sigma^2 = \sigma^2(1 + c_i)$ when the noise is regarded as independent of the smooth. From this, an unbiased estimated estimator of σ^2 is given by $\hat{\sigma}^2 = n^{-1} \sum_i (R_i - \hat{\alpha}_i)^2/(1 + c_i)$. Consequently, since a posteriori $\hat{\alpha}$ has a normal distribution with mean α, the approximate 95% credibility intervals for α are given by $\hat{\alpha} \pm 1.96 S.E.(\hat{\alpha})$.

The above methods were tested on the data from an acid rain, wet-deposition time series and very encouraging results have been obtained. In the case considered in WZ, only locally constant, linear and quadratic (i.e., $p \leq 2$) structural models are considered for illustration.

In general, the WZ-approach has some advantages over the other nonparametric smoothing methods decribed in section 2. Firstly, the methods mentioned in section 2 share the shortcoming that the explicit form of the estimates of $s(\cdot)$, especially for extrapolants, and confidence bands are hard to derive. For example, Silverman (1985) points out that in spline smoothing only implicit solution can be obtained except for large samples when an approximate explicit expression may be derived. In contrast, the WZ-approach does yield explicit estimates of $s(\cdot)$ with corresponding explicit credibility bands.

Secondly, the prior that Wahba (1983) uses is essentially unique so that the applicability of the spline smoothing is severely limited. The exponential smoothing method suffers from the lack of an objective procedure for model identification. In contrast, in the WZ-approach, every aspect of the analysis is guided by the local-structural model and the Bayesian paradigm itself. Prior knowledge may be incorporated directly into the analysis and "borrowing from strength" is feasible.

The "backfitting" method for estimating the smoothing parameter c is simple and computationally fast compared with other procedures. Furthermore, the WZ-approach can be extended to the analysis of multiple, multivariate space-time series and the regression problem. Weerahandi and Zidek (1986) have successfully extended the WZ-approach to the analysis of multiple independent time series. In the latter case, exact credibility bands for growth curves are generated. Their work on spatio-temporal analysis is in progress and some promising unpublished preliminary results have been obtained.

3.2 Estimating the "smoothing parameter" c for locally regular structural models.

(I) <u>Cross-validation method</u>

In the case considered in WZ, $\hat{c}$ is found by minimizing $XV(c)$ which depends on σ^2 only through c. The results of WZ suggest that quite satisfactory estimates of c are obtained in the locally constant, linear and quadratic structural models. So, it is tempting to try the same approach to estimate $c = \hat{c}$ for the other pth order locally regular structural models when S is assumed to have $p+1$ derivatives.

It is then necessary to find out if $\hat{c}$ is really the global minimum, rather than a local minimum of the function $XV(c)$. Two ways which help to determine this are:

(i) Plot $XV(c)$ against c and examine the shape of $XV(c)$ for large c. In this case, we need to know the asymptotic behaviour of $XV(c)$ as $c \to \infty$.

Let

$t_1 < ... < t_n$,

$t_1 < ... < t_n$, $x_j = t_j - t_i$ for all $j = 1, ..., n$ excluding $j = i$,

$\hat{\alpha}_{(i)}(c) =$ the cross-validated estimate of α at t_i,

$XV(\infty) =$ value of $XV(c)$ at $c = \infty$, and

an accent on $\sum$ will denote summation over all $j = 1, ..., n$ excluding $j = i$.

In the case of a locally constant fit (i.e. $p = 0$), $\hat{\alpha}_{(i)}(c)$ can be obtained by

the generalized least squares procedure described in section 3.1. It is given by:

$$\hat{\alpha}_{(i)}(c) = \frac{\sum' r_j (1 + cx_j^2)^{-1}}{\sum' (1 + cx_j^2)^{-1}}.$$

It follows that

$$\hat{\alpha}_{(i)}(c) \to \frac{\sum' r_j x_j^{-2}}{\sum' x_j^{-2}} \stackrel{\text{def}}{=} \hat{\alpha}_{(i)}(\infty) \qquad \text{as} \quad c \to \infty.$$

Therefore

$$\begin{aligned} XV(\infty) &\approx \sum_{i=1}^{n} \left(r_i - \hat{\alpha}_{(i)}(\infty) \right)^2 \\ &\approx \sum_{i=1}^{n} \left(r_i - \frac{\sum' r_i x_j^{-2}}{\sum' x_j^{-2}} \right)^2. \end{aligned}$$

Similarly, in the case of a locally linear fit (i.e., $p = 1$),

let

$$z_j = (1 + cx_j^4)^{-1},$$

then

$$\hat{\alpha}_{(i)}(c) = \frac{\sum' x_j^2 z_j \sum' r_j z_j - \sum' x_j z_j \sum' x_j r_j z_j}{\sum' z_j \sum' x_j^2 z_j - (\sum' x_j z_j)^2}.$$

It follows that

$$\hat{\alpha}_{(i)}(c) \to \frac{\sum' x_j^{-2} \sum' r_j x_j^{-4} - \sum' x_j^{-3} \sum' r_j x_j^{-3}}{\sum' x_j^{-4} \sum' x_j^{-2} - (\sum' x_j^{-3})^2} \stackrel{\text{def}}{=} \hat{\alpha}_{(i)}(\infty) \quad \text{as} \quad c \to \infty.$$

Therefore

$$\begin{aligned} XV(\infty) &\approx \sum_{i=1}^{n} \left(r_i - \hat{\alpha}_{(i)}(\infty) \right)^2 \\ &\approx \sum_{i=1}^{n} \left(r_i - \frac{\sum' x_j^{-2} \sum' r_j x_j^{-4} - \sum' x_j^{-3} \sum' r_j x_j^{-3}}{\sum' x_j^{-4} \sum' x_j^{-2} - (\sum' x_j^{-3})^2} \right)^2. \end{aligned}$$

By using the same technique, we can find $XV(\infty)$ for higher order locally regular structural models.

(ii) Examine the asymptotic behaviour of $XV(c)$ for large c. If $XV(c)$ is less than $XV(\infty)$ for large c, then we will have increased confidence that there is not a local minimum at infinity.

In the locally constant case:

$$\begin{aligned} r_i - \hat{\alpha}_{(i)}(c) &= r_i - \frac{\sum' r_j (1+cx_j^2)^{-1}}{\sum' (1+cx_j^2)^{-1}} \\ &\approx r_i - \frac{\sum' \left(r_j c^{-1} x_j^{-2}(1-c^{-1}x_j^{-2})\right)}{\sum' \left(c^{-1}x_j^{-2}(1-c^{-1}x_j^{-2})\right)} \qquad \text{for large } c \\ &= r_i - \frac{\sum' r_j x_j^{-2} - c^{-1}\sum' r_j x_j^{-4}}{\sum' x_j^{-2}\left(1 - c^{-1}\frac{\sum' x_j^{-4}}{\sum' x_j^{-2}}\right)} \\ &\approx r_i - \frac{\left(\sum' r_j x_j^{-2} - c^{-1}\sum' r_j x_j^{-4}\right)\left(1 + c^{-1}\frac{\sum' x_j^{-4}}{\sum' x_j^{-2}}\right)}{\sum' x_j^{-2}} \qquad \text{for large } c. \end{aligned}$$

Let

$$\frac{1}{\sum' x_j^{-2}} = a_i, \quad \sum' r_j x_j^{-2} = b_i, \quad \sum' r_j x_j^{-4} = d_i, \quad \text{and} \quad \frac{\sum' x_j^{-4}}{\sum' x_j^{-2}} = e_i.$$

Then

$$\begin{aligned} r_i - \hat{\alpha}_{(i)}(c) &= r_i - a_i\left(b_i - \frac{d_i}{c}\right)\left(1 + \frac{e_i}{c}\right) \\ &= r_i - a_i\left(b_i + \frac{1}{c}(b_i e_i - d_i) - \frac{1}{c^2}(d_i e_i)\right) \\ &\approx r_i - a_i b_i - \frac{1}{c}a_i(b_i e_i - d_i) \qquad \text{for large } c \\ &= r_i - \hat{\alpha}_{(i)}(\infty) + \frac{f_i}{c} \end{aligned}$$

where

$$f_i = -a_i(b_i e_i - d_i).$$

So

$$XV(c) = \sum_i (r_i - \hat{\alpha}_{(i)}(c))^2$$

$$= \sum_i \left((r_i - \hat{\alpha}_{(i)}(\infty)) + \frac{f_i}{c} \right)^2 \qquad (3.2.1).$$

$$\approx \sum_i (r_i - \hat{\alpha}_{(i)}(\infty))^2 + \frac{2}{c} f_i (r_i - \hat{\alpha}_{(i)}(\infty))$$

Differentiating (3.2.1) with respect to c, we obtain

$$XV(c)' = -\frac{2}{c^2} f_i (r_i - \hat{\alpha}_{(i)}(\infty))$$

$$= \frac{2}{c^2} \sum_i \left(\frac{\sum' r_j x_j^{-2}}{\sum' x_j^{-4}} - \frac{\sum' r_j x_j^{-4}}{\sum' x_j^{-2}} \right) \left(r_i - \frac{\sum' r_j x_j^{-2}}{\sum' x_j^{-2}} \right) \qquad \text{for large } c.$$

Similarly, by using the same technique, we can estimate $XV(c)'$ for higher order locally regular structural models.

(II) <u>"Backfitting" method</u>

Recall from equation (3.1.1), that if S has $p+1$ derivatives, then the a priori model is of the form

$$R_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + ... + \beta_p x_i^p + \beta_{p+1} x_i^{p+1} + \epsilon \qquad (3.2.2)$$

where $\beta_0 = S(t_{n+1})$, $\beta_i = D^i S(t_{n+1})$, $x_i = (t_i - t_{n+1})^i / i!$, $i = 1, ..., p+1$ and $\epsilon = \eta + N$ as defined in section 3.1.

Recall that δ^2 represents the uncertainty about the size of $D^{p+1}S/(p+1)!$ in the remainder term of the Taylor expansion of S i.e. δ^2 is the a priori variance of $D^{p+1}S/(p+1)!$. Hence $t_{n+1} = t_i$ may be chosen successively , and δ^2 may be

estimated by

$$\hat{\delta}^2 = \text{ sample variance of } (\hat{\beta}_{p+1}/(p+1)!)$$
$$= \frac{\text{sample variance of } (\hat{\beta}_{p+1})}{((p+1)!)^2}.$$

We also note that $c = \delta^2/\sigma^2$, so the value of c for the order p, c_p, say, is given by

$$\begin{aligned} c_p &\approx \frac{\delta^2}{\sigma^2} \\ &= \frac{\text{sample variance of } (\hat{\beta}_{p+1})}{\hat{\sigma}^2((p+1)!)^2} \\ &= \frac{(n-1)^{-1}\sum_{i=1}^{n}(\hat{\beta}_{p+1}(i) - \bar{\beta}_{p+1})^2}{\hat{\sigma}^2((p+1)!)^2} \end{aligned} \tag{3.2.3}$$

where $\hat{\beta}_{p+1}(i)$ is the coefficient of x_i^{p+1}, $\bar{\beta}_{p+1}$ is the sample mean of $\hat{\beta}_{p+1}$'s, and $\hat{\sigma}^2$ can be found by the generalized least squares procedure described in section 3.1.

Thus, if (3.2.2) holds, by the same argument, we can use (3.2.3) to estimate the values of c_j for any order $j = 0, ..., p$.

However, to use the formula (3.2.3) we need c_{p+1}. We may use the cross-validatory approach mentioned in section 3.1 to estimate c_{p+1}, from which we can obtain estimates of β_i, δ^2 and σ^2. Alternatively, we may simply put $c_{p+1} = 0$ assuming p is large. The reason is that at a certain stage, the uncertainty about the size of $D^{p+1}S/(p+1)!$ will be reduced approximately to zero for sufficiently large p. So, it yields the result $\delta^2 \to 0$, hence $\hat{c}_{p+1} \to 0$.

In summary, if (3.2.2) holds and c_{p+1} is fixed, we may estimate the c_j of the lower orders $j \leq p$ by using (3.2.3). We refer to such an approach as the "backfitting" procedure.

(III) **Conclusions**

Comparing methods (I) and (II), we prefer the "backfitting" approach to cross-validation for the following reasons. Firstly, it is computationally more efficient since we do not need to compute $\hat{r}_i$ for each p and c, and we can estimate $\hat{c}_j$ for all $j \leq p$ within one run once $\hat{c}_{p+1}$ is fixed. Secondly, this approach coheres with the theory proposed in section 3.1 and we expect the resulting curves are comparatively smoother. In consequence, the smoothers obtained for different orders of local regularity become comparable.

3.3 Choosing the "optimal" order p.

For many practical purposes, it is sufficient to choose p subjectively. Once $\hat{c}_p$ is determined, say by either of the methods decribed in section 3.2, we may plot the resulting curves and then choose the one which "looks best". However, in many contexts it is desirable to have an automatic procedure for choosing p. Hence, one of our main objectives in this thesis is to find the "optimal" value of p in the sense that $\hat{r}$, estimated by the pth order locally regular fit, minimizes some measure of accuracy. In our case, we use the predictive squared error PSE (to be defined later).

A fit giving a minimal PSE yields the model of the data which gives the best prediction (in the least square sense) of each data point. In other words, the minimum PSE gives the parameters which maximize the internal consistency of the data set with respect to the estimated model. Our procedure is as follows:

1. Fix the span of observations m ($m < n$).
2. Fix the value of $p+1$ and set $\hat{c}_{p+1} = 0$.

3. Use the first m observations and apply the "backfitting" procedure described in section 3.2 to estimate $\hat{c}_0$, $\hat{c}_1$, ... , $\hat{c}_j$, for $j = 0, ..., p$.

4. Once $\hat{c}_j$ is found, apply the generalized least squares procedure described in section 3.1 to estimate $r_{m+1} = \hat{r}_{m+1}(\hat{c}_j)$ for each $j \leq p$.

5. Shift to the next m observations i.e. $r_2, ..., r_{m+1}$ and repeat steps 3 and 4 to estimate $\hat{r}_{m+2}(\hat{c}_j)$ for each $j \leq p$.

6. Repeat step 5 for successive m observations until $\hat{r}_n(\hat{c}_j)$ for each $j \leq p$ is found.

7. Add the sum of squared deviations from steps 4 to 6 for each $j \leq p$. This is the predictive squared error,

$$PSE(j) \stackrel{\text{def}}{=} \frac{1}{n-m} \sum_{i=1}^{n-m} (r_{m+i} - \hat{r}_{m+i}(\hat{c}_j))^2, \quad j = 0, ..., p.$$

8. Compare the $PSE(j)$ for each $j \leq p$ to find the optimal $j \leq p$ which minimizes the PSE.

In the procedure decribed above, we are predicting one observation ahead each time. It is perhaps more natural to predict the the average of w observations ahead since $S(t_{n+1})$, the quantity of interest, is the smooth, i.e. the process stripped of noise at time $t = t_{n+1}$. Our intention can be easily realized by a slight modification of the above procedure. It is done as follows:

(i) In step 4, use the $\hat{c}_j$ to find $\hat{r}_{m+1}(\hat{c}_j), ..., \hat{r}_{m+w}(\hat{c}_j)$ for each $j \leq p$.

Let

$$\bar{r}_i(\hat{c}_j) = \frac{1}{w} \sum_{k=1}^{w} \hat{r}_{m+k+i-1}(\hat{c}_j).$$

(ii) In step 6, repeat step 5 for successive m observations until $\bar{r}_{w'}(\hat{c}_j)$ ($w' =$

$n - m - w + 1$), for each $j \leq p$ are found.

(iii) In step 7, for each $j \leq p$, replace $PSE(j)$ by

$$\frac{1}{w'} \sum_{i=1}^{w'} (\bar{r}_i - \bar{r}_i(\hat{c}_j))^2$$

where

$$\bar{r}_i = \frac{1}{w} \sum_{k=1}^{w} r_{m+k+i-1}.$$

Finally, follow the same steps 1 to 8 in the original procedure with the above modifications. In section 4 , both types of predictions are considered for illustrations.

4. APPLICATION.

4.1 Introduction

In this section, the methods described in the last section are tested on the data used in WZ. The data were obtained from one of the nine stations of the MAP3S/PCN monitoring network which is found in the Northeastern part of the United States and has provided event-based chemical measurements of wet deposition events since about 1976. A detailed description of the data set can be found in Gentleman, Zidek and Olsen (1985). For illustrative purposes, our analyses are confined to two subcases: (i) the field pH values which were measured at a station located in Pennsylvania State University during the precipitation events of 1977 (80 observations were recorded), and (ii) the monthly average pH-values over the 1976-82 period at the same station (71 observations were recorded). Hereafter (i) is referred to as the pH77 data and (ii) as the M.A. pH data. Throughout this section time is measured in days starting with January 1, 1976 which is regarded as the origin.

Recall that in equation (3.1.1), $c_p = \delta^2/\sigma^2$. However, δ^2, the variance of $\beta_{p+1}/(p+1)!$, is extremely small. So $w_i = t_i - t_{n+1}$ is rescaled by a factor 10^{-2}, $i = 1, ..., n$, in estimating c_p numerically. In this way, all the $\{\hat{c}_j\}$ obtained are enlarged by a factor of 10^{4j+4} for all $j = 0, ..., p$. The results reported below are obtained by the methods described in section 3; all the calculations were performed by using Fortran subroutines which are included in the Appendix.

4.2 Comparison of the cross-validation and the backfitting methods.

Before making any comparisons of these two methods, we examined the asymptotic behaviour of $XV(c)$ for large c, to confirm that the cross-validatory estimate of c_p being found is really the global minimum of $XV(c)$. Only the locally constant fit is considered as an illustration. Observe in Table I that the $XV(c)$ values increase as c tends to infinity for both the pH77 and the M.A. pH data. Also, for large c, $XV(c)' \approx 0.2738c^{-2}$ and $0.000372c^{-2}$ for the pH77 and the M.A. pH data respectively. Therefore, we are sure that the global minimum of $XV(c)$, $\hat{c}_p$, is finite.

In applications, $\hat{c}_p$, the cross-validatory value of c_p, is found by using an IMSL (1982) optimization subroutine to minimize $XV(c)$. The estimated values of $\hat{c}_p$ for the pH77 data are presented in Table II and the corresponding locally polynomial fits are displayed in Figure 1(A-C). Figure 1 shows that the locally quadratic fit is smoother than the other polynomial fits in the sense that it cannot make the quick turns so it tends to round off the corners of the dips of the data. This may due to the fact that $\hat{c}_2$ is comparatively small so a greater degree of smoothness is revealed in the resulting curve. As we have pointed out in section 3.1 δ^2 is decreasing as p is increasing, so $\hat{c}_p$ should be decreasing too. However, as p increases, $\hat{c}_p$ become relatively large and the resulting curves look quite wiggly. Nevertheless, $\hat{c}_{10} = 0$ is obtained. It is noteworthy that unlike what happens in regression, the fit here is found to be fairly robust with respect to the degree of the polynomial within the range of the data.

When $\hat{c}_{10} = 0$, we can apply the backfitting procedure descibed in section 3.2 to obtain $\tilde{c}_p$, the backfitted values of c_p, $p = 0, ..., 9$. The results are summarized in Table II and the corresponding locally polynomial fits are portrayed in

TABLE I: Estimated values of XV(c) for a locally constant fit to the pH77 and M.A. pH data.

C	XV(c) (pH77)	XV(c) (M.A. pH)
0	6.0485	3.5161
3.2 *	4.5944	3.0132
128.0 **	3.8339	3.2376
1.0E+3	4.0742	3.2611
1.0E+4	4.5993	3.2642
1.0E+5	4.7845	3.2644
1.0E+6	4.8083	3.2646
1.0E+7	4.8108	3.2647
1.0E+8	4.8110	3.2648
1.0E+9	4.8111	3.2648
1.0E+10	4.8111	3.2648
∞	4.8112	3.2648

* cross-validatory estimate of c for the M.A. pH data

** cross-validatory estimate of c for the pH77 data

TABLE II: Estimated values of Cp for the pH77 data.

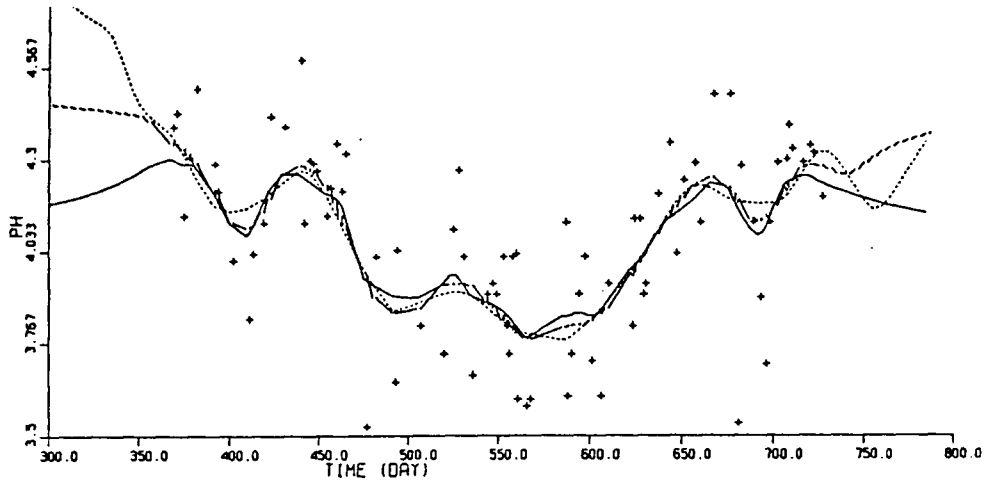
Order (p)	$\hat{C}p$ *	$\tilde{C}p$ **
0	128	45
1	1750	4034
2	1028	75721
3	350000	279977
4	1715000	452421
5	2486000	171752
6	616000	47119
7	46300	1989
8	4700	107
9	400	.002
10	0	

* $\hat{C}p$: Cp obtained by the cross-validation method

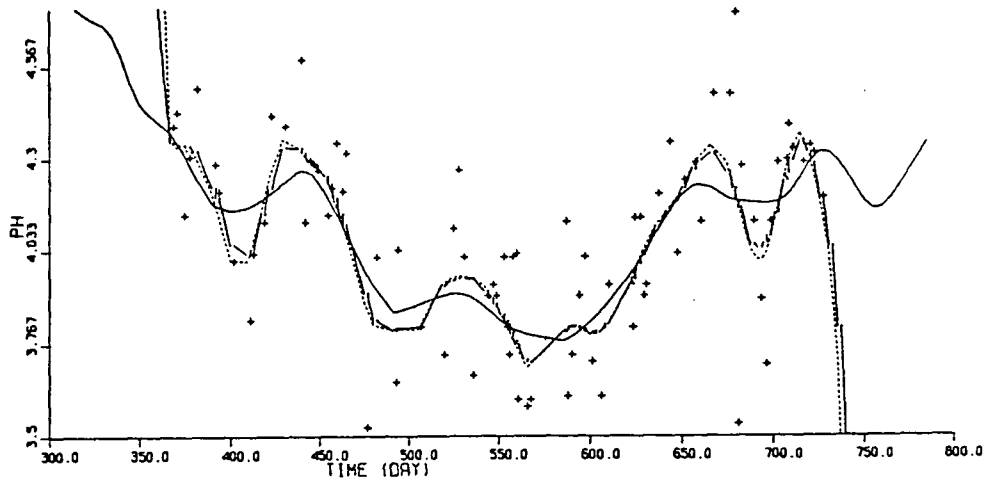
** $\tilde{C}p$: Cp obtained by the backfitting method

FIG. 1: A comparison of the locally polynomial fits of different order p to the pH77 data. (using the cross-validation method)

A.
p=0 (solid line)
p=1 (broken line)
p=2 (dotted line)



B.
p=2 (solid line)
p=3 (broken line)
p=4 (dotted line)



C.
p=4 (solid line)
p=7 (broken line)
p=10(dotted line)

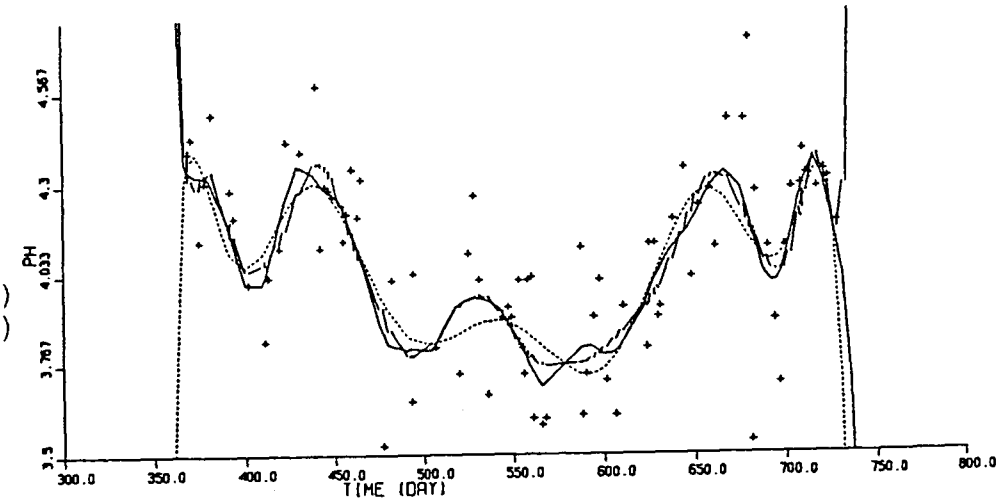
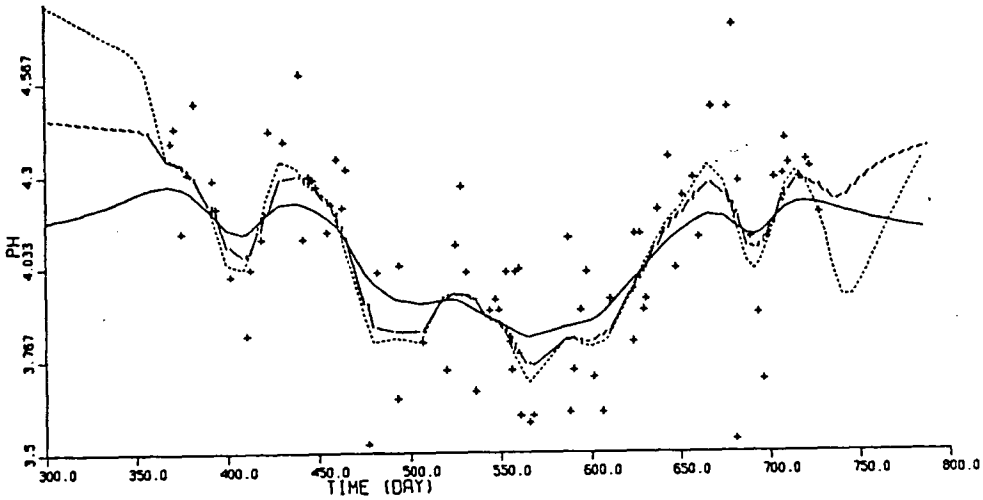
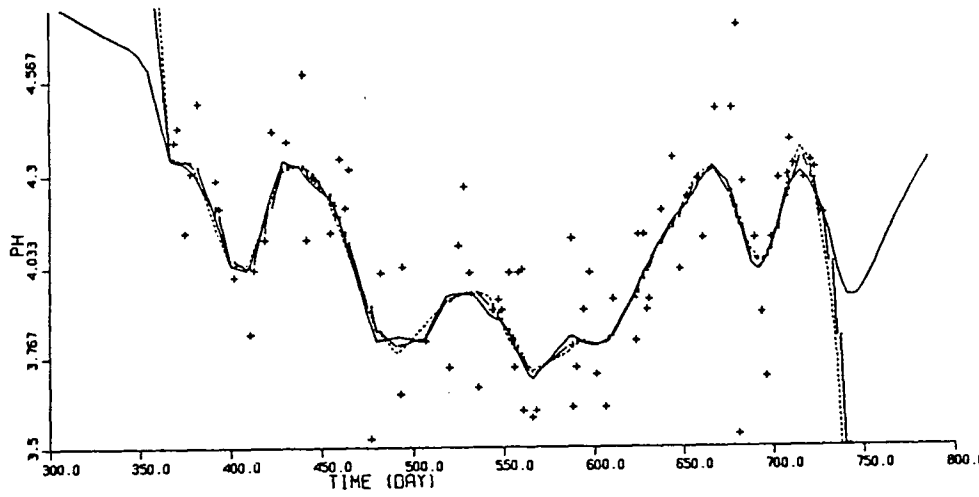


FIG. 2: A comparison of the locally polynomial fits of different order p to the pH77 data. (using the backfitting method)

A.
p=0 (solid line)
p=1 (broken line)
p=2 (dotted line)



B.
p=2 (solid line)
p=3 (broken line)
p=4 (dotted line)



C.
p=4 (solid line)
p=7 (broken line)
p=9 (dotted line)

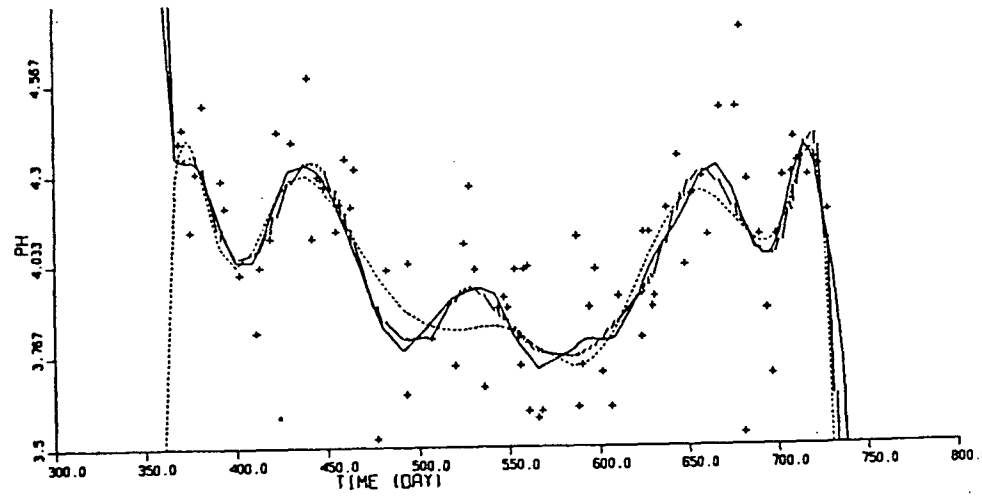


Figure 2(A-C). In general, the estimated values of $\tilde{c}_p$ are comparatively smaller than the corresponding $\hat{c}_p$ except in the cases of locally linear and quadratic fits. The locally constant and quadratic fits obtained by these two methods are compared in Figure 3. The resulting curves do not change drastically unless the difference between $\hat{c}_p$ and $\tilde{c}_p$ is greater than a factor of 10.

As a final note about Figures 1 and 2, the locally polynomial fits emerge from the data at either extremity with a marked upward or downward trend as the order p increases. This is inevitable; as WZ point out , for large $|t_{n+1}|$, the α_i's become the least squares polynomial estimates.

For for the M.A. pH data, the estimated values of $\hat{c}_p$ and $\tilde{c}_p$ are given in Table III and the corresponding locally polynomial fits are displayed in Figure 4(A-C) and 5(A-C). Since the estimated $\tilde{c}_p$'s are relatively smaller, the corresponding locally polynomial fits obtained by the backfitting method are comparatively smoother than those obtained by the cross-validation method. In Figure 6 are contrasted the results of the locally constant and quartic fits obtained by these two methods. However, little difference is observed.

The credibility intervals for the locally quadratic and cubic fits are portrayed in Figure 7. We must be careful in interpreting those credibility intervals. The bands in Figure 7 are not simultaneous interval estimates. Those in Figure 7 merely indicate the pointwise credibility interval for $S(t_{n+1})$ at each t_{n+1}. It is clear that the bands are blowing up at either end of the data. This is consistent with the unsatisfactory predictions revealed in Figures 1 and 2. Using the backfitting method, similar results are obtained though they are not presented here.

FIG 3: A comparison of the locally constant fits (top) and quadratic fits (bottom) using the cross-validation method (solid line) and the backfitting method (dotted line) for the pH77 data.

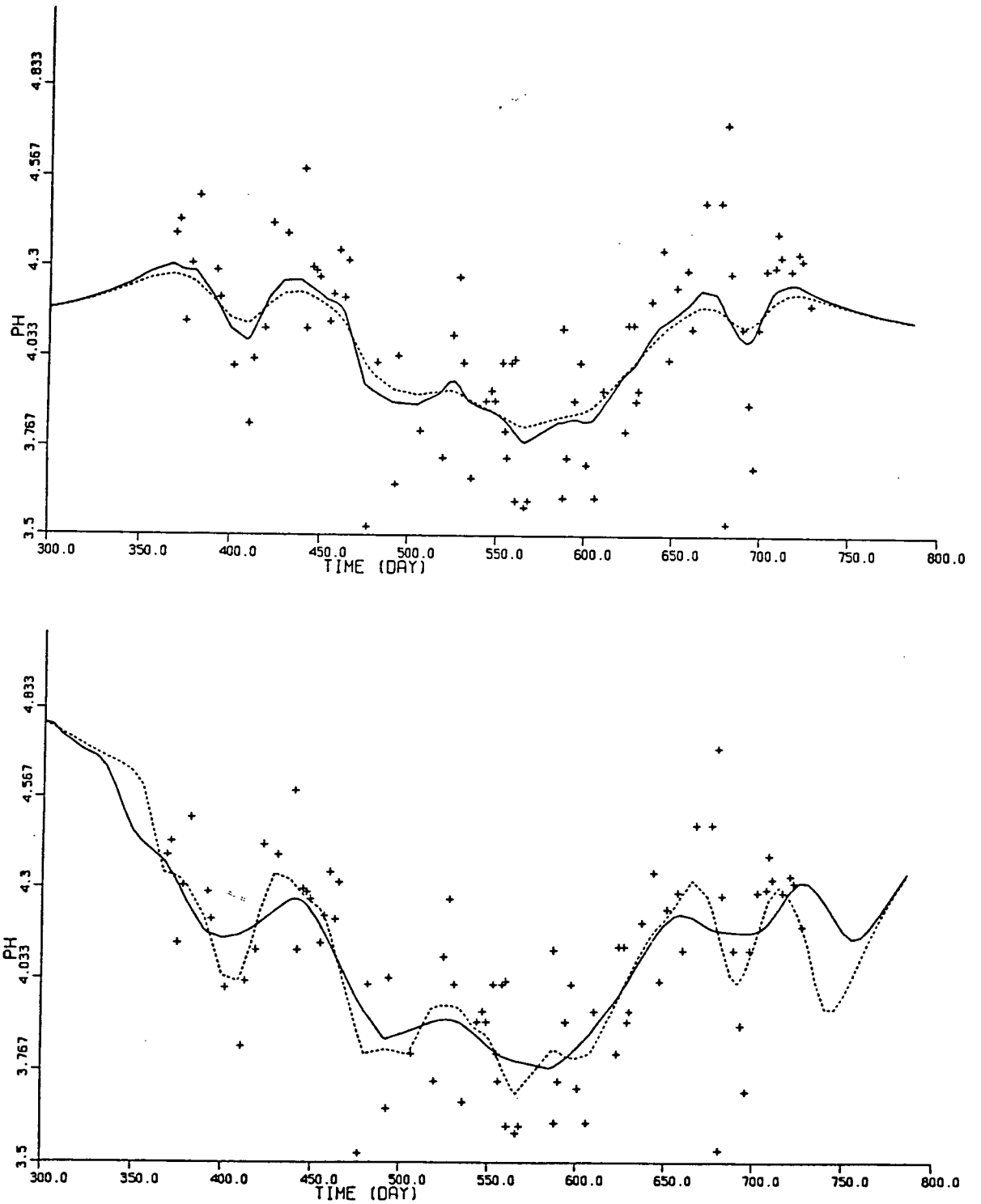


TABLE III: Estimated values of Cp for the M.A. pH data.

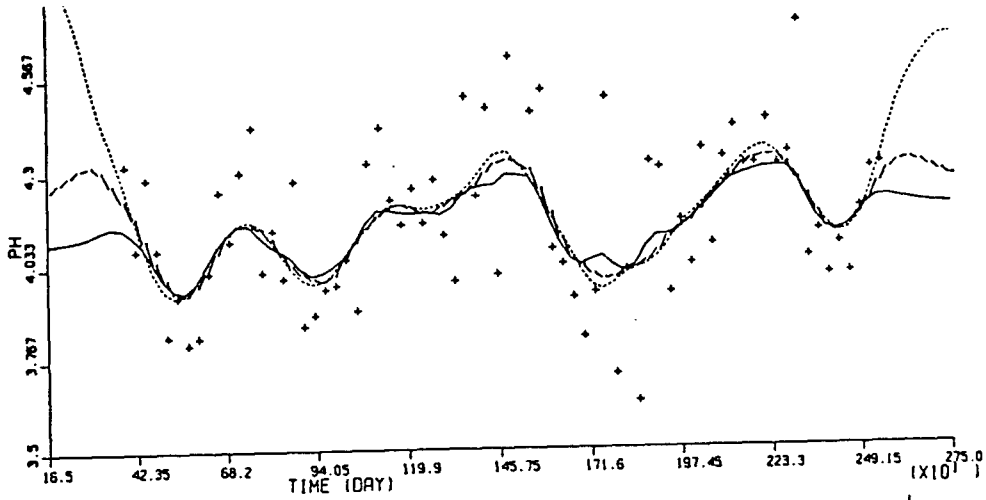
Order (p)	$\hat{Cp}$ *	$\tilde{Cp}$ **
0	3.20629	.94601
1	1.65739	.39114
2	.17786	.02716
3	.06700	.00179
4	.08250	.00018
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0
10	0	

* $\hat{Cp}$: Cp obtained by the cross-validation method

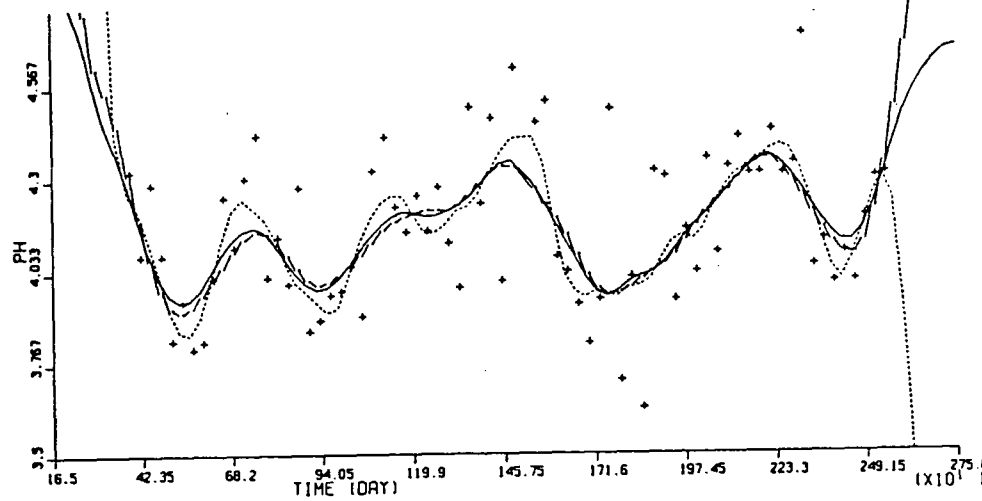
** $\tilde{Cp}$: Cp obtained by the backfitting method

FIG. 4: A comparison of the locally polynomial fits of different order p to the M.A. pH data. (using the cross-validation method)

A.
p=0 (solid line)
p=1 (broken line)
p=2 (dotted line)



B.
p=2 (solid line)
p=3 (broken line)
p=4 (dotted line)



C.
p=4 (solid line)
p=7 (broken line)
p=10(dotted line)

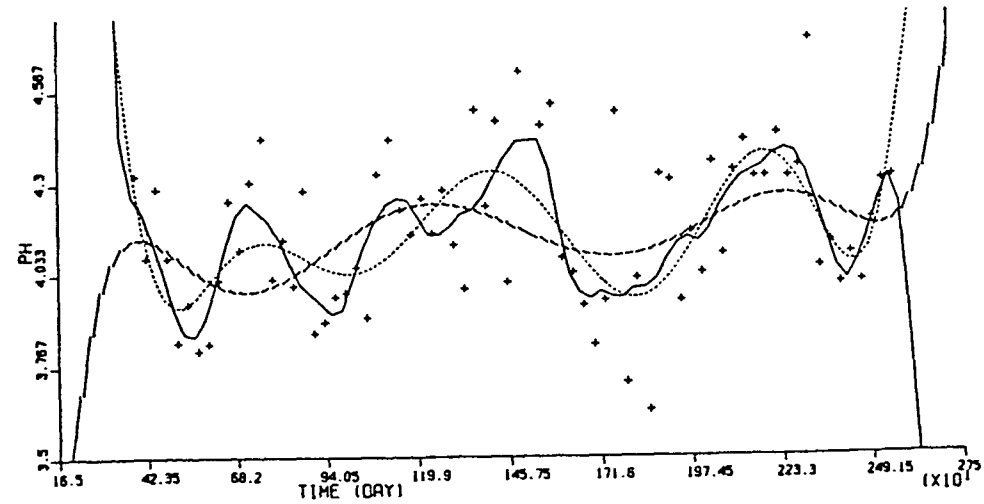
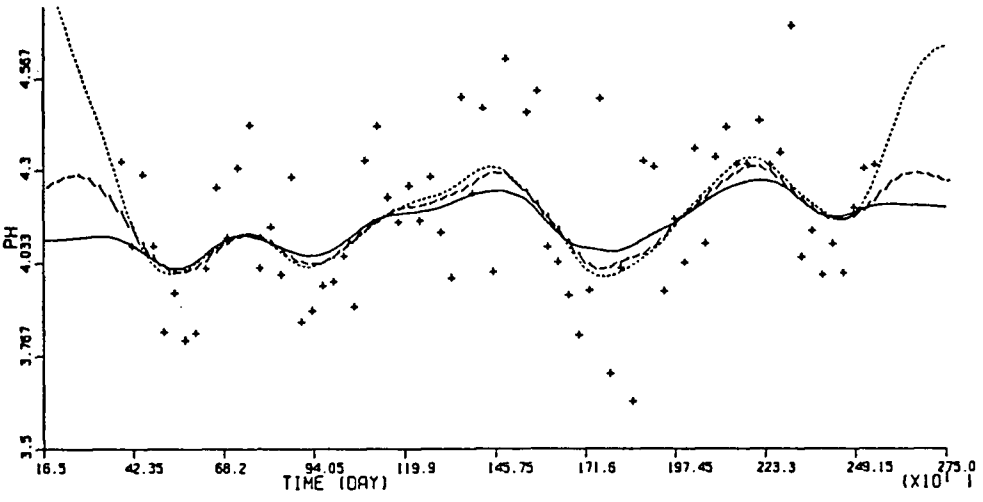
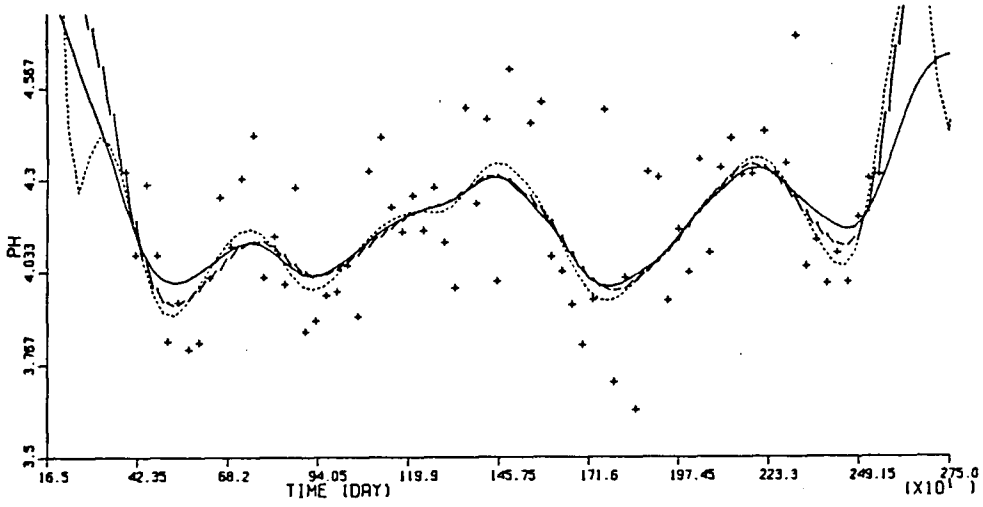


FIG. 5: A comparison of the locally polynomial fits of different order p to the M.A. pH data. (using the backfitting method)

A.
p=0 (solid line)
p=1 (broken line)
p=2 (dotted line)



B.
p=2 (solid line)
p=3 (broken line)
p=4 (dotted line)



C.
p=4 (solid line)
p=7 (broken line)
p=9 (dotted line)

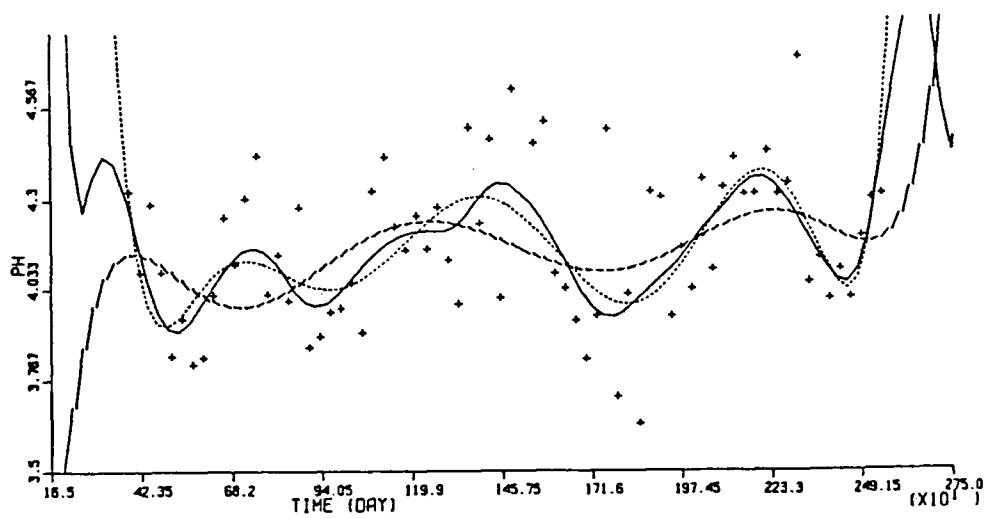
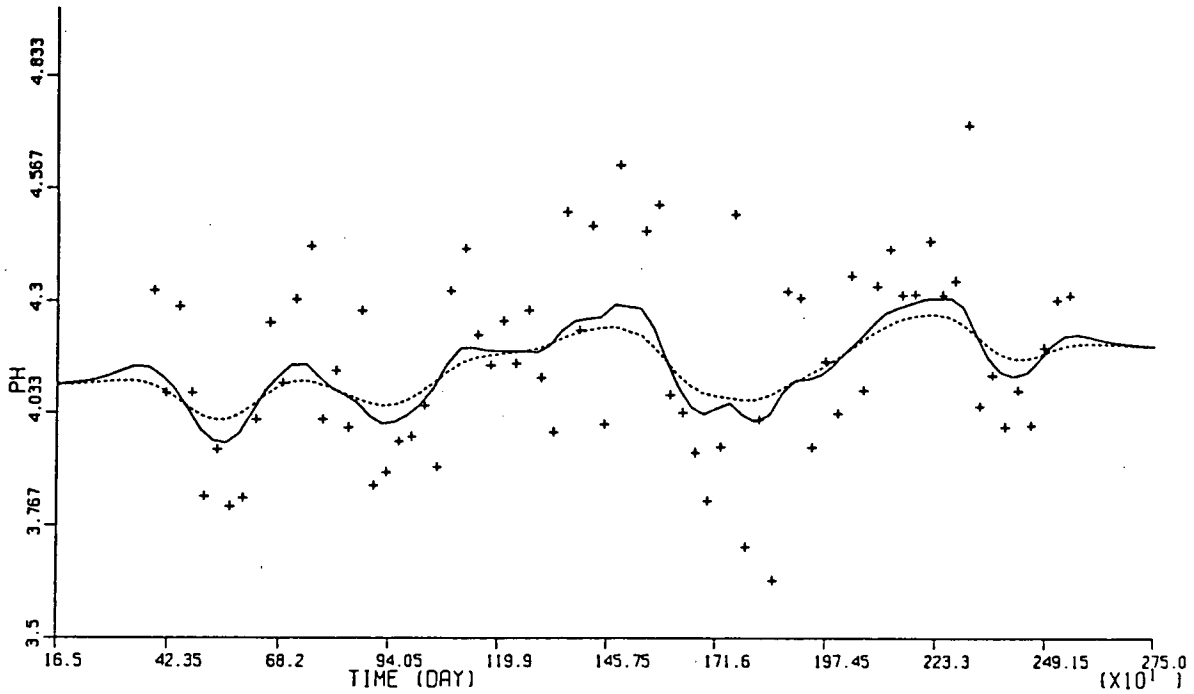


FIG 6: A comparison of the locally constant fits (top) and quartic fits (bottom) using the cross-validation method (solid line) and the backfitting method (dotted line) for the M.A. pH data.



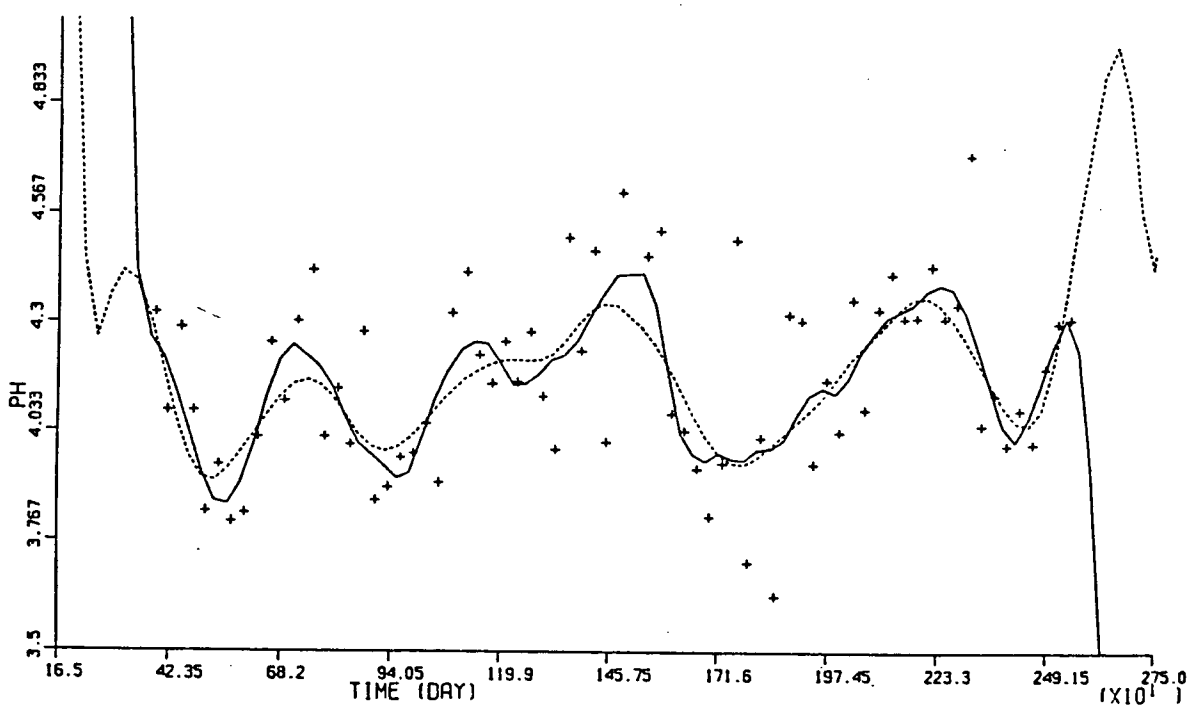
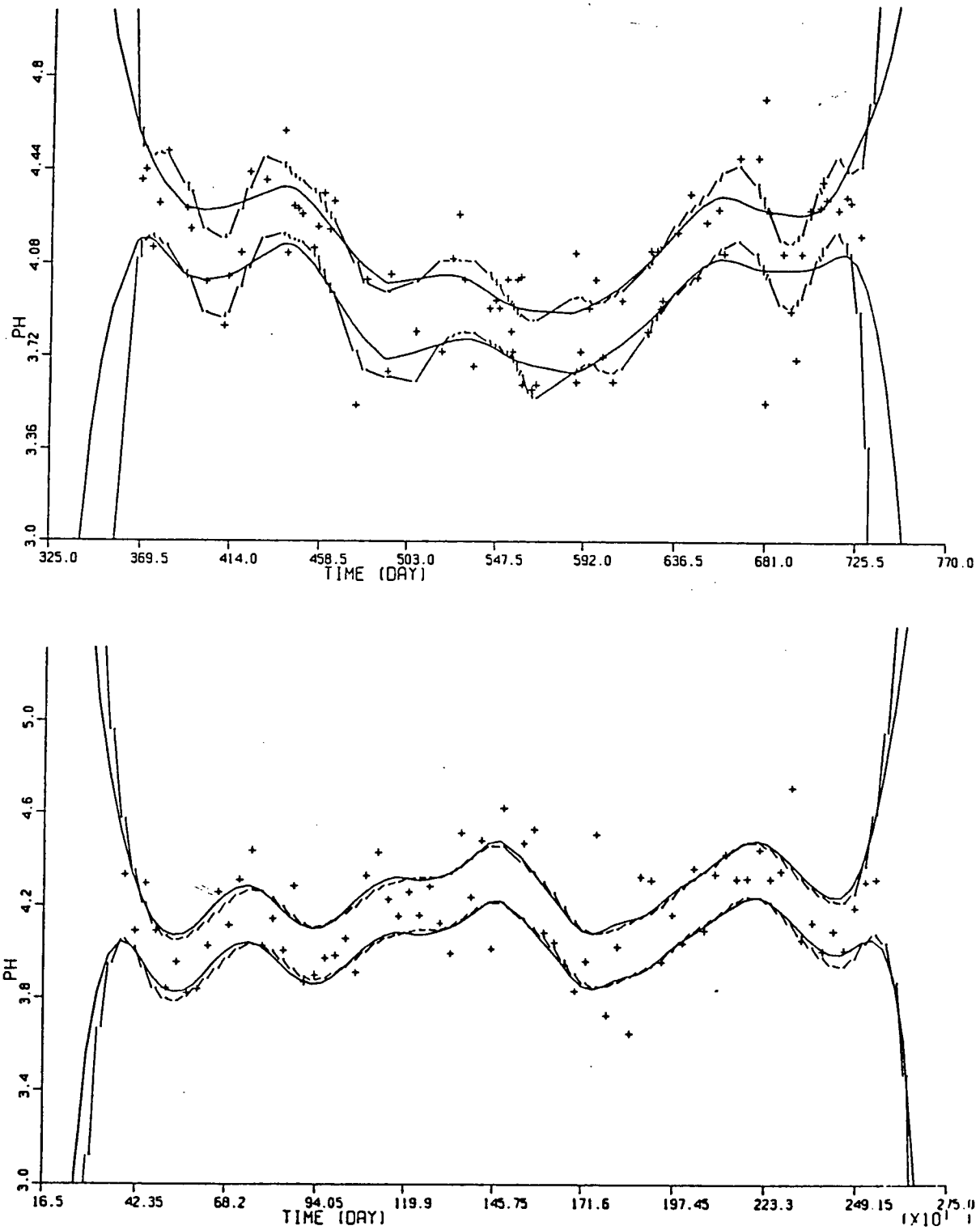


FIG 7: A comparison of the credibility intervals for the locally quadratic (solid line) and cubic (broken line) fits for the pH77 (top) and the M.A. pH (bottom) data. (using the cross-validation method)



Some differences between our approach and classical parametric methods, e.g., least squares, are obvious. Our approach is more flexible since we can adjust the smoothing parameter c subjectively to get a "nice looking" curve. The choice $\hat{c} = 0$ gives the least squares polynomial fit. For example, when $p = 1$, we get a least squares line passing through the data; however, the underlying pattern cannot be recaptured. Moreover, it is easy to generate the locally polynomial fit of any order p with corresponding pointwise credibility intervals. One more interesting feature is shown in Fig. 4 of WZ where the least squares linear and locally linear fits are contrasted when the middle of the pH77 data are deleted. The width of the confidence band for the least squares linear fit reaches a minimum where data are missing. However, the width of the credibility interval for the locally linear fit increases in the gap in accordance with our expectations.

4.3 Results of choosing the "optimal" order p.

Applying the procedure described in section 3.3, the estimated values of $PSE(p)$ for the pH77 data are obtained and presented in Tables IV and V. The results clearly indicate that the optimal order is $p = 0$ since the locally constant fit attains the minimum PSE for all of the spans of observations, m, we used. Because the data are not monotone and the noise is significant, the best fit is simply obtained by taking the weighted average of the observations in the span.

At the same time, we have compared our results with the simplest smoother, the moving mean, where $\hat{r}_i$ is estimated by the average of the observations in the span. The comparisons are reported in Tables IV and V. Although the

TABLE IV: Estimated values of PSE(p) for the pH77 data.
(predict one observation ahead)

p \ m	15	20	30	40	50	60	70
0	.0738	.0773	.0639	.0615	.0718	.0851	.0393
1	.1298	.1305	.1009	.0906	.0795	.0986	.0492
2	.3804	.3484	.2300	.1657	.1470	.1540	.0586
3	5.089	3.719	.5573	.4074	.3671	.2573	.0533
4	39.08	19.21	1.764	1.330	.7226	.3866	.0709
5	381.6	102.8	7.905	5.760	1.795	.4719	.1126
6	1768.8	461.8	18.21	22.68	1.994	.5999	.1643
7	63212	3159.3	46.04	27.64	2.818	.7892	.2380
8	2769231	81920	378.2	31.22	4.956	.9524	.2700
9	22131	30632	68.36	5.792	1.021	.4001	.1977
moving mean	.0673	.0762	.0848	.1026	.0940	.1064	.0678

TABLE V: Estimated values of PSE(p) for the pH77 data.
(predict the average of 10 observations ahead)

p \ m	15	20	30	40	50	60
0	.0335	.0328	.0289	.0311	.0387	.0123
1	.1250	.1004	.0693	.0746	.0822	.0919
2	1.491	1.105	.7047	.5908	.5588	.7217
3	38.67	18.06	9.541	8.317	6.819	8.919
4	1176.4	430.9	137.4	113.7	63.56	64.48
5	39996	9323.5	1938.8	1408.3	537.2	333.4
6	1.4E+6	2.5E+5	37510	19741	4108.5	1504.1
7	7.9E+7	5.4E+6	6.3E+5	2.2E+5	18543	3038.3
8	1.2E+9	7.7E+8	9.5E+6	1.2E+6	75790	5949.6
9	3.5E+9	2.4E+10	2.4E+6	25596	1841.4	362.1
moving mean	.0428	.0507	.0582	.0497	.0342	.0194

moving mean has a simple model formulation, it is not capable of giving a better prediction than the locally constant fit. For the M.A. pH data, there is smaller variation in the pH values over the range of time, and again the locally constant fit is best though the results are not shown here.

We have tested the efficacy of this procedure on two other data sets.

Data set 1:

We apply the procedure to data on the annual catch of lobsters by Maine fishermen taken from Morrison (1983) (abbreviated hereafter by Lobster data). The data consists of the number of registered lobsters traps used by Maine fishermen over the years 1932-75. The correlation coefficient between these two variables is close to 0.9 and this strongly suggests a linear dependence of the number of traps on time. This is supported by the results given in Table VI. The locally constant, linear and quadratic fits are compared in Figure 8A. These three curves are quite similar to each other even though the locally linear fit minimizes the PSE. The credibility intervals of the locally linear fit are portrayed in Figure 8B. As we have pointed out above such bands are only pointwise interval estimates.

Data set 2:

One hundred equally spaced points were generated from the sinusoid $y = 2\cos x - 5\sin x + 6$ on the interval [0,10]. We apply the same procedure to the first 80 points and then compare the prediction of the last 20 points by using different locally polynomial fits. As the results show in Table VII, PSE is minimized by different values of $j \leq p$ depending on the size of span m. For interpolation, all the locally polynomial fits appear to pass through the data points equally

TABLE VI: Estimated values of PSE(p) for the Lobster data--- predict one observation ahead (top) and predict the average of 10 observations ahead (bottom).

p \ m	20	30
0	3.4 E + 4	5.8 E + 4
1	3.3 E + 4	5.6 E + 4
2	8.1 E + 4	1.3 E + 5
3	2.0 E + 5	3.1 E + 5
4	4.9 E + 5	7.1 E + 5
5	1.1 E + 6	1.5 E + 6
6	2.6 E + 6	2.7 E + 6
7	6.4 E + 6	3.6 E + 6
8	1.7 E + 7	3.3 E + 6
9	2.8 E + 6	1.6 E + 6
moving mean	1.7 E + 5	3.5 E + 5

p \ m	20	30
0	7.7 E + 4	1.4 E + 5
1	1.3 E + 4	2.2 E + 4
2	6.7 E + 4	5.1 E + 4
3	4.3 E + 5	2.3 E + 5
4	4.0 E + 6	1.6 E + 6
5	4.7 E + 7	1.2 E + 7
6	6.1 E + 8	9.5 E + 7
7	9.4 E + 9	1.4 E + 9
8	1.7 E +11	2.5 E +10
9	4.9 E +12	4.8 E +10
moving mean	1.5 E + 5	3.3 E + 5

FIG 8A: A comparison of the locally constant (broken line), linear (solid line) and quadratic (dotted line) fits for the Lobster data. (using the backfitting method)

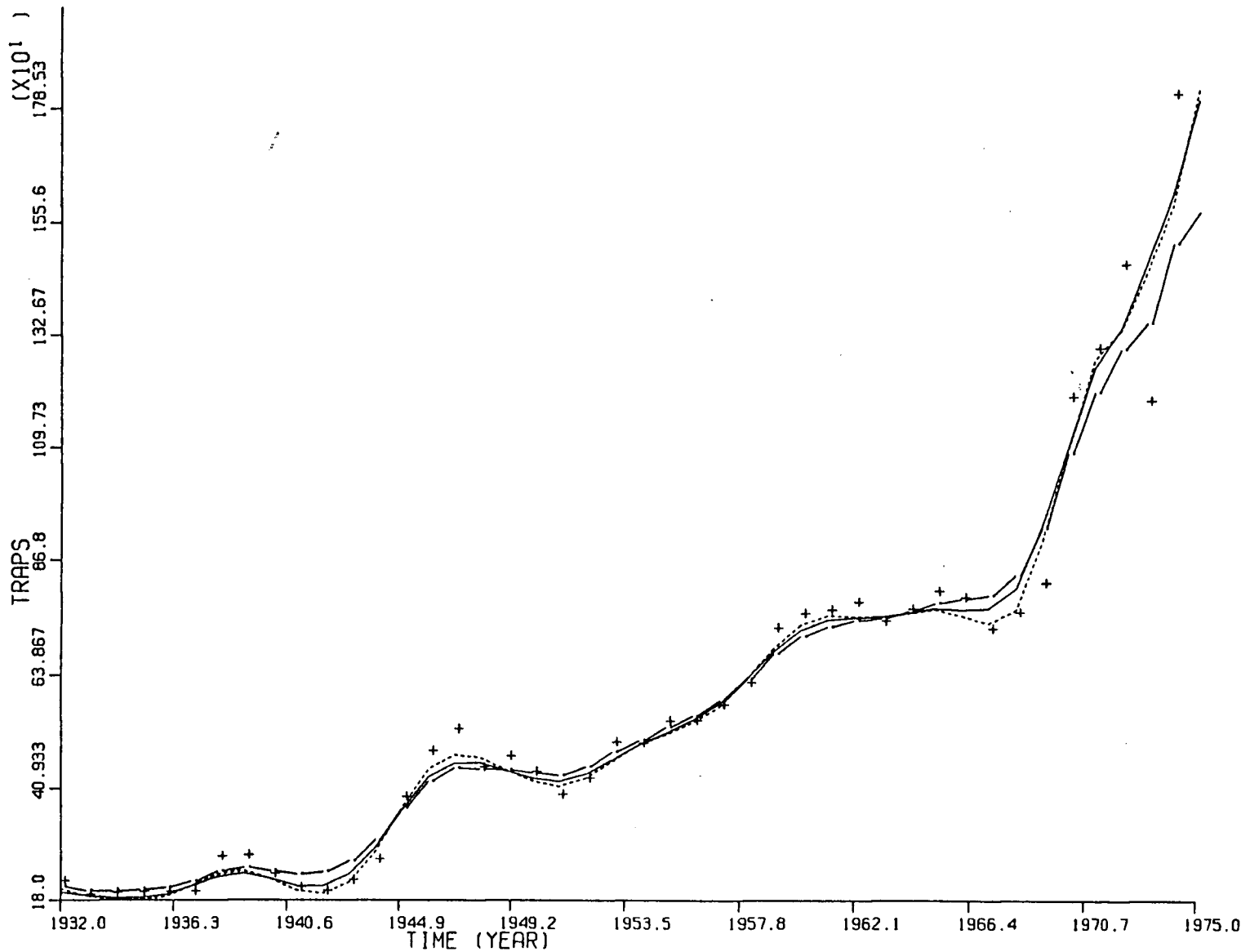


FIG 8B: Locally linear fit with the credibility intervals for the Lobster data. (using the backfitting method)

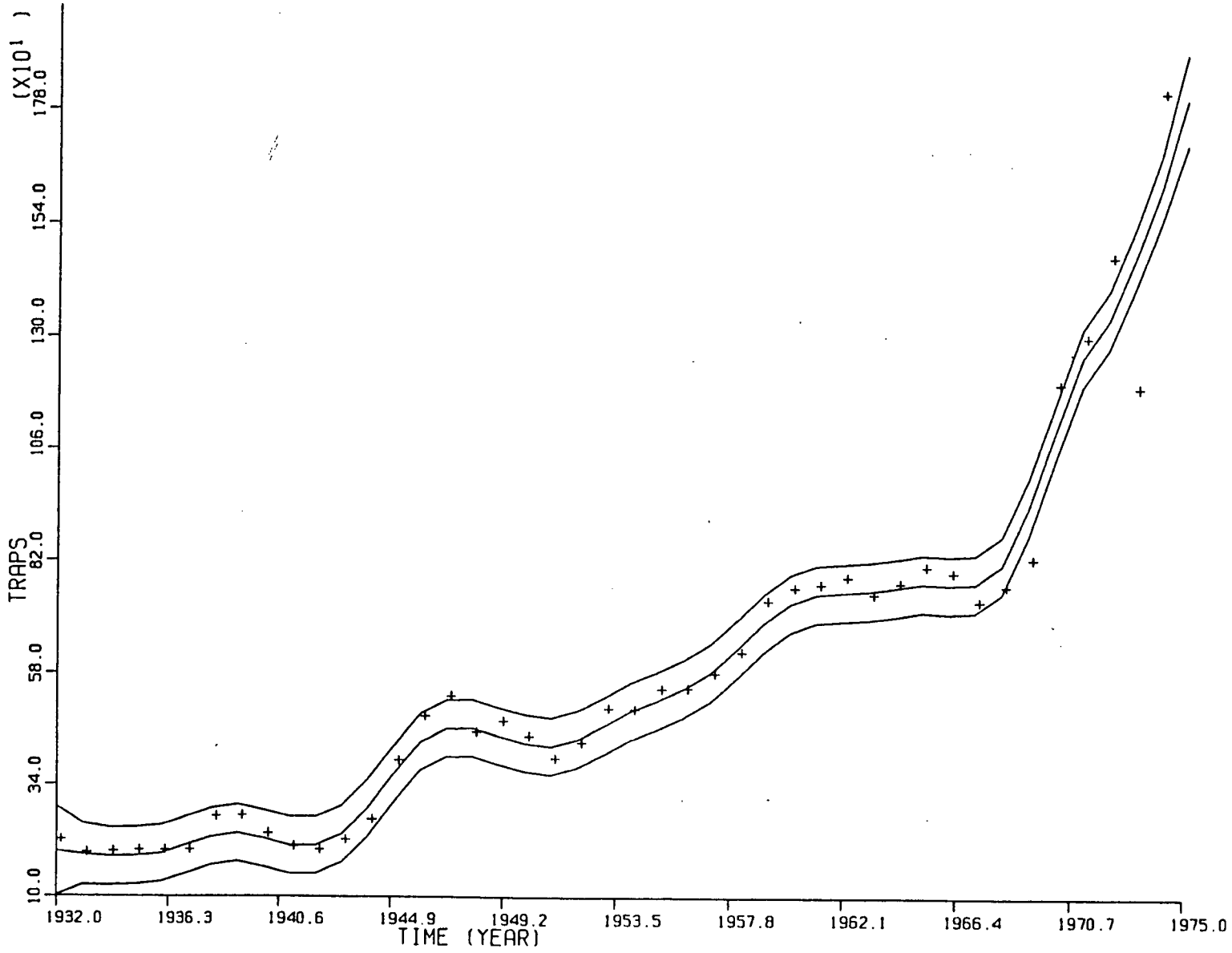


TABLE VII: Estimated values of PSE(p) for the sinusoid ($y = 3\cos x - 5\sin x + 6$)--predict one observation ahead (top) and predict the average of 10 observations ahead (bottom).

p \ m	20	40	60
0	7.8 E - 1	9.8 E - 1	1.1 E 0
1	1.5 E - 2	1.8 E - 2	1.6 E - 2
2	2.4 E - 4	5.2 E - 4	6.4 E - 4
3	3.3 E - 6	6.6 E - 6	1.0 E - 5
4	4.3 E - 8	1.7 E - 7	2.4 E - 7
5	2.0 E - 9	3.2 E - 9	1.0 E - 8
6	4.5 E - 9	1.5 E - 9	1.0 E - 9
7	1.4 E - 8	1.3 E - 9	6.0 E -10
8	4.2 E - 8	1.3 E - 9	4.0 E -10
9	1.5 E - 8	4.0 E -10	1.2 E - 9

p \ m	20	40	60
0	1.2 E + 1	1.2 E + 1	1.2 E + 1
1	4.6 E 0	6.3 E 0	9.3 E 0
2	6.8 E - 1	1.7 E - 0	6.3 E - 1
3	1.0 E - 1	3.0 E - 1	9.0 E - 1
4	6.5 E - 3	3.7 E - 2	1.5 E - 2
5	5.6 E - 4	3.4 E - 3	1.6 E - 2
6	4.0 E - 5	2.4 E - 4	1.8 E - 4
7	2.9 E - 4	5.0 E - 5	1.1 E - 4
8	1.9 E - 2	3.5 E - 4	3.0 E - 5
9	1.6 E + 6	9.0 E - 5	1.0 E - 5

well since no noise exists in the data. These situations are illustrated in Figures 9(A-B). Figure 9B indicates that locally the 9th order polynomial gives the best prediction of the last 20 data points.

From these two examples, we can see that the proposed procedure is capable of selecting the locally polynomial fit which captures the underlying pattern of the data.

To conclude this section, it should be added that we have examined the behaviour of the yearly differences of the M.A. pH data over the range of time, $D(t_{n+1}) = R(t_{n+1}) - R(t_{n+1} - \text{ 1 year })$. This is done to see if any seasonality existed over the years. Figure 10 is a plot of the locally constant fit with credibility intervals. No apparent seasonality is observed in the picture. However, we can see that $D(t_{n+1})$ is increasing and decreasing in alternating years. Such a yearly cycle of the change of pH is clearly shown. Of course, related subject knowledge would be needed to explain such a phenomenon.

FIG 9A: A comparison of the locally linear (broken line), 5th order polynomial (solid line) and 10th order polynomial (dotted line) fits for the sinusoid ($y = 3\cos x - 5\sin x + 6$) data. (using the cross-validation method)

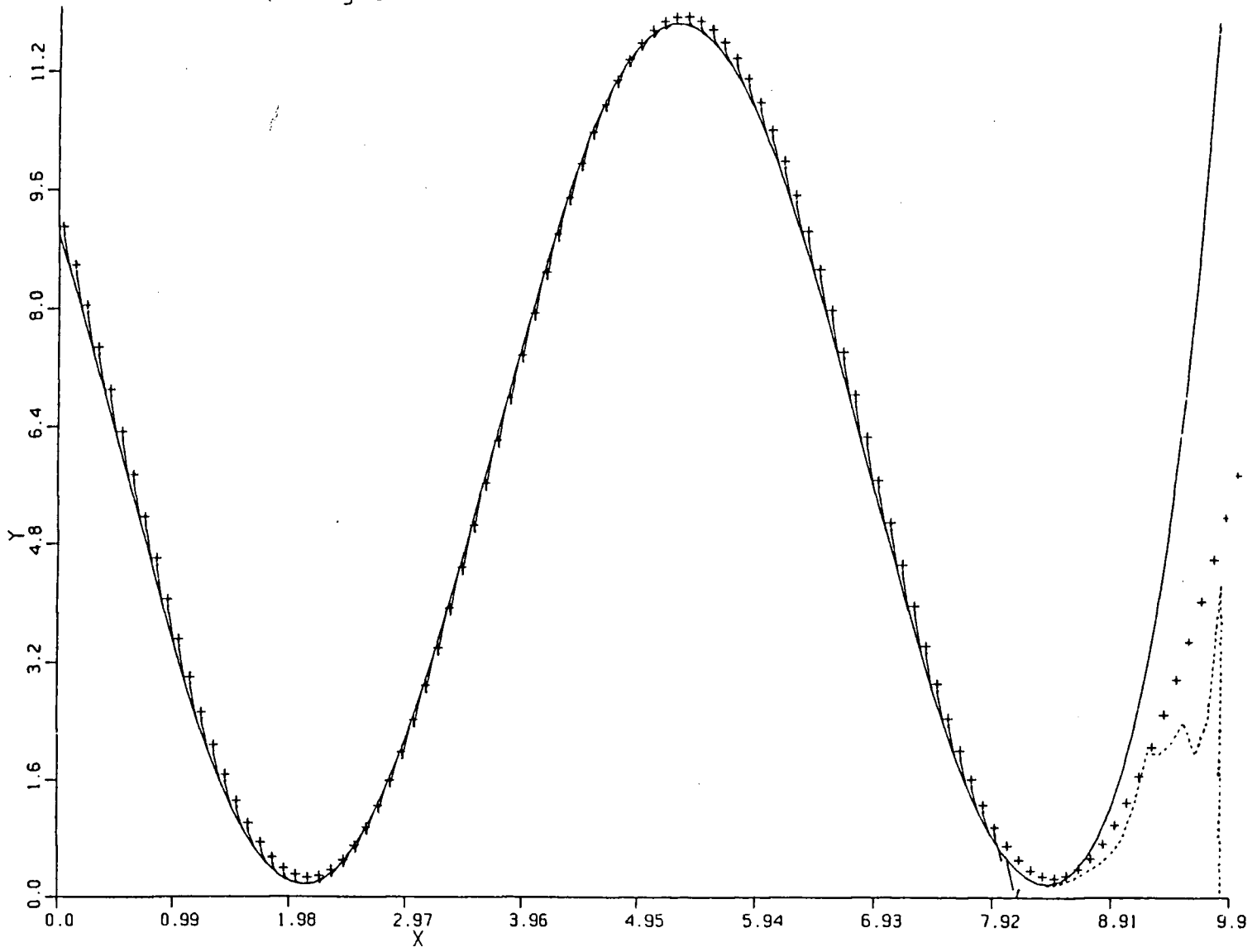


FIG 9B: A comparison of the locally 5th order polynomial (solid line), 7th order polynomial (broken line) and 9th order polynomial (dotted line) fits for the sinusoid (y = 3cosx - 5sinx + 6) data. (using the backfitting method)

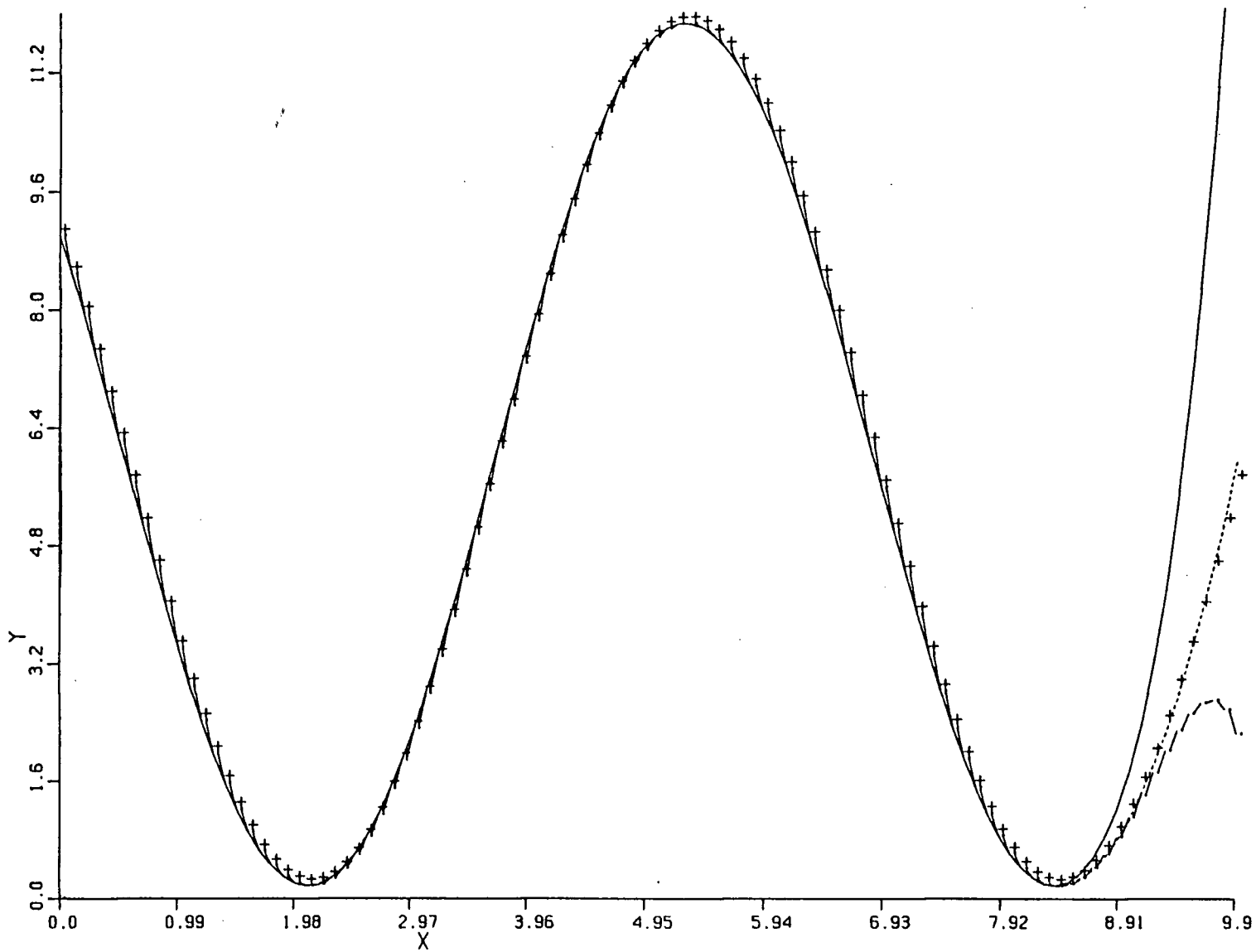
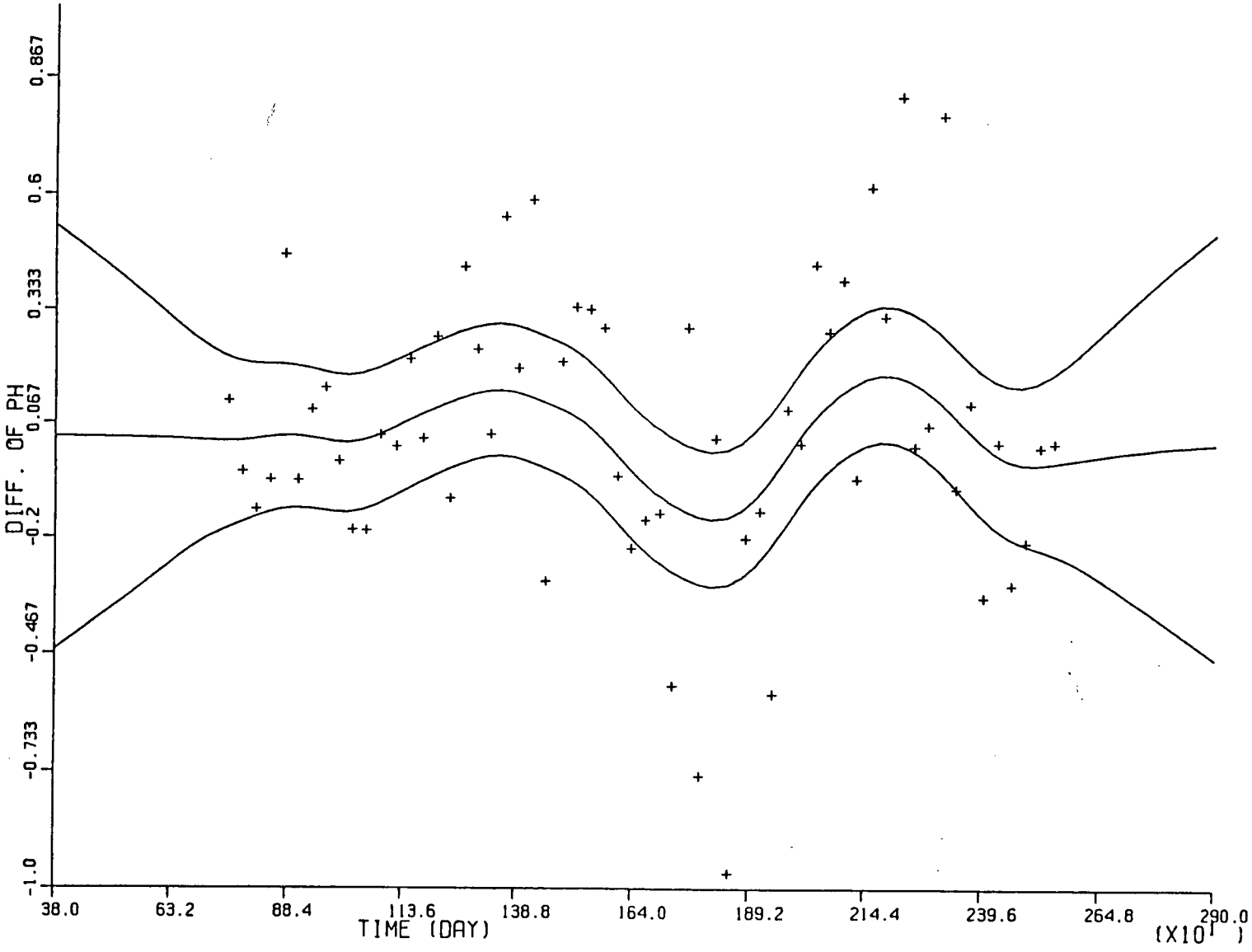


FIG 10: Locally constant fit with the credibility intervals
for the difference of the M.A. pH data.
(using the backfitting method)



5. CONCLUSIONS.

In section 2, we survey some of the popular nonparametric smoothing methods. A brief comparison of these methods with the WZ-approach is made.

As mentioned in section 3.2, through the examination of the asymptotic behaviour of $XV(c)$, we can be sure that the cross-validatory estimate of c being found is finite. Unfortunately, the cross-validatory estimates of c are relatively large and the resulting curves are wiggly. The problem is that we allow the data "speak freely for itself" but neglect the formulated model. As an alternative, the backfitting method is more coherent with the theory in section 3.1 and the resulting curves are comparatively smoother in the sense that less turns are generated. Also, it is computationally fast and we can estimate $\hat{c}_j$ for all $j \leq p$ within one run once $\hat{c}_{p+1}$ is fixed.

In section 4, the efficacy of a data-based procedure was demonstrated for choosing the "optimal" order p. Satisfactory results are achieved in two other examples too. It would be interesting to repeat these tests on data generated by a smooth function plus noise to see if the smooth function would be recovered.

In practice, we can arbitrarily set $c_{p+1} = 0$ for some high order p and then apply the procedure to choose the appropriate order of locally polynomial fit. Once the order is determined, we can use the backfitting estimate of c to generate the interpolants or predictands with corresponding pointwise credibility intervals. Nevertheless, as we have pointed out above the fit is fairly robust with respect to the order of the polynomial, a slight difference of the order p would not change the appearance of the resulting curve very much.

On the whole, the proposed approach is flexible and simple to implement. In general, it is easy to generate the locally polynomial fit of any order if it is needed and appropriate degree of smoothness can be obtained through the backfitting method. Moreover, this approach does provide explicit estimates which are much different in form to those derived by classical methods.

In conclusion, we suggest that further research be carried out, including

1. a proper Bayesian analysis for controlling the extrapolation of $S(t)$, and
2. the development of the theory for multivariate space-time series and the regression problem.

BIBLIOGRAPHY

Becker, R. A. and Chambers, J. M. (1984): *S: An Interactive Environment for Data Analysis and Graphics.* California: Wadsworth.

Bickel, P. J. and Rosenblatt, M. (1973): On some global measures of the density function estimates. *Ann. Statist.* **1** 1071-1095.

Bierens, H. J. (1983): Sample moments integrating normal kernel estimators of multivariate density and regression functions. *Sankhya B* **45** 160-192.

Brown, R. G. and Meyer, R. F. (1961): The fundamental theorem of exponential smoothing. *Operations Research* **9** 673-685.

Clark, R. M. (1977): Nonparametric estimation of a smooth regression function. *J. R. Statist. Soc. B* **39** 107-113.

Cleveland, W. S. (1979): Robust locally weighted regression and smoothing scatterplots. *J. Amer. Statist. Assoc.* **74** 829-836.

Collomb, G. (1981): Estimation nonparametrique de la regression: revue bibliographique. *Inter. Statist. Review* **49** 75-93.

Craven, P. and Wahba, G. (1979): Smoothing noisy data with spline functions. *Numer. Math.* **31** 377-403.

Dubrule, O. (1983): Two methods with different objectives: splines and kriging. *Math. Geology* **15** 245-257.

Dubrule, O. (1984): Comparing splines and kriging. *Computers and Geosciences* **10** 327-338.

Enns, P. G., Machak, J. A., Spivey, W. A., and Wrobleski, W. J. (1982): Forecasting applications of an adaptive multiple exponential smoothing model. *Management Science* **28** 1035-1044.

Friedman, J. H. (1984): A variable span smoother. Stanford University LCS Technical Report No. **5**.

Gardner, E. S. (1985): Exponential smoothing: the state of art. *J. Forecasting* **4** 1-28.

Gentleman, R., Zidek, J. and Olsen, R. (1985): MAP3S/PCN acid rain precipitation monitoring data base. University of British Columbia Statistics Department Technical Report No. **19**.

Hardle, W. and Marron, J. S. (1985): Optimal bandwidth election in non-parametric regression function estimation. *Ann. Statist.* **13** 1465-1481.

Harvey, A. C. (1984): Analysis and generalization of a multivariate exponential smoothing model. *LSE Econometrics Programme Discussion Paper No.* **A.41**.

Huber, P. J. (1979): Robust smoothing. In *Robustness in Statistics.* (R. L. Launer and G. N. wilkinson, eds.) New York: Academic Press.

IMSL (1980): *International Mathematical and Statistics Libraries, Inc., Sub-*

routine **ICSSCV.**

IMSL (1982): *International Mathematical and Statistics Libraries, Inc., Subroutine* **ZXLSF.**

Jones, R. H. (1966): Exponential smoothing for multivariate times series. *J. R. Statist. Soc. B* **34** 385-392.

Lenth, R. V. (1977): Robust splines. *Comm. in Statist.-A* **6** 847-854.

Lindley, D. V. and Smith, A. F. M. (1972): Bayes estimates for the linear model. *J. R. Statist. Soc. B* **34** 1-18.

Makridakis, S., Andersen, A., Carbone, R., Fildes, R., Hibon, M., Newton, J., Lewandowski, R., Parzen, R. and Winkler, R. (1982): The accuracy of extrapolation (time series) methods: results of a forcasting competition. *J. of Forscasting* **1** 111-153.

Makridakis, S. and Wheelwright, S. C. (1978): *Interactive Forecasting: Univariate and Multivariate Methods* (2nd ed.). San Francisco: Holden-Day.

Morrison, D.F. (1983): *Applied Linear Statistical Methods.* Prentice-Hall.

Nadaraya, E. A. (1964): On estimating regression. *Theory Probab. Appl.* **9** 141-142.

Priestley, M. B. and Chao, M. T. (1972): Nonparametric function fitting. *J. R. Statist. Soc. B* **34** 385-392.

Reinsch, C. H. (1967): Smoothing by spline functions. *Numer. Math.* **10** 177-183.

Rosenblatt, M. (1956): Remarks on some nonparametric estimates of a density function. *Ann. Math. Statist.* **27** 832-837.

Rosenblatt, M. (1971): Curve estimates. *Ann. Math. Statist.* **42** 1815-1842.

Schoenberg, I. J. (1964): Spline functions and the problem of graduation. *Proc. Nat. Acad. Sci. USA* **52** 947-950.

Schuster, E. F. (1972): Joint asymptotic distribution of the estimated regression function at a finite number of distinct points. *Ann. Math. Statist.* **43** 84-88.

Silverman, B. V. (1982): Kernel density estimation using the fast Fourier transform. *Appl. Statist.* **31** 93-99.

Silverman, B. V. (1984): A fast and efficient cross-validation method for smoothing parameter choice in spline regression. *J. Amer. Statist. Assoc.* **79** 584-589.

Silverman, B. V. (1985): Some aspects of the spline smoothing approach to nonparametric regression curve fitting. *J. R. Statist. Soc. B* **47** 1-21.

Stone, C. J. (1977): Consistent nonparametric regression. *Ann. Statist.* **5** 595-645.

Stone, M. (1974): Cross-validatory choice and assessment of statistical predictions. *J. R. Statist. Soc. B* **36** 111-133.

Trigg, D. W. and Leach, A. G. (1967): Exponential smoothing with an adaptive response rate. *Opl. Res. Q.* **18** 53-63.

UBC MATRIX (1982): A guide to solving matrix problems. University of British Columbia Computing Documentation.

Utreras, F. I. (1979): Cross-validation techniques for smoothing spline functions in one or two dimensions. In *Smoothing Techniques For Curve Estimation.* (T. Gasser and M. Rosenblatt, eds.) Lecture Notes in Mathematics, No. **757.** Springer-Verlag, Berlin.

Utreras, F. I. (1981): On computing robust splines and applications. *SIAM J. Sci. Stat. Comput.* **2** 153-163.

Wahba, G. (1980): Spline bases, regularization, and generalized cross-cross-validation for solving approximation problems with large quantities of noisy data. In *Approximation Theory III* (E. W. Cheney ed.) New York: Academic Press.

Wahba, G. (1983): Bayesian "confidence intervals" for the cross-validation smoothing spline. *J. R. Statist. Soc. B* **45** 133-150.

Wahba, G. (1984a): Cross-validated spline methods for the estimation of multivariate functions from data on functionals. In *Statistics: An Appraisal.* (H. A. David and H. T. David eds.) The Iowa State University Press.

Wahba, G. (1984b): Surface fitting with scattered noisy data on Euclidean d-space and on the sphere. *Rocky Mountain Journal of Mathematics* **14** 281-299.

Wahba, G. and Wendelberger, J. (1980): Some new mathematical methods for variational objective analysis using splines and cross-validation. *Monthly Weather Review* **108** 1122-1143.

Wahba, G. and Wold, S. (1975): A completely automatic french curve: fitting spline functions by cross-validation. *Comm. in Statist.* **4** 1-17.

Watson, G. S. (1964): Smooth regression analysis. *Sankhya A* **26** 359-372.

Watson, G. S. (1984): Smoothing and interpolation by kriging and with splines. *Math. Geology* **16** 601-615.

Weerahandi, S. and Zidek, J. (1985): Smoothing locally smooth processes by Bayesian nonparametric methods. University of British Columbia Statistics Department Technical Report No. **26**.

Weerahandi, S. and Zidek, J. (1986): Analysis of multiple time series by Bayesian and Empirical Bayesian nonparametric methods. University of British Columbia Statistics Department Technical Report No. **33**.

Wegman, E. J. (1980): Two approaches to nonparametric regression: splines and isotonic inference. In *Recent Developments in Statistical Inference and Data Analysis* (K. Matusita ed.). North-Holland.

Wegman, E. J. and Wright, I. W. (1983): Splines in statistics. *J. Amer.*

Statist. Assoc. **78** 351-365.

Whittaker, E. T. (1923): On a new method of graduation. *Proc. Edinburgh Math. Soc.* **41** 63-75.

Winters, P. R. (1960): Forscasting sales by exponentially weighted moving averages. *Management Sci.* 324-342.

APPENDIX

This appendix contains Fortran subprograms for the computations of (i) the smoothing parameter c by cross-validation, (ii) the smoothing parameter c by "backfitting", (iii) the smooth curve estimates with corresponding credibility intervals, and (iv) the values of PSE.

```
      DOUBLE PRECISION FUNCTION F(C)
C
C     THIS IS A ROUTINE FOR COMPUTING THE SMOOTHING
C     PARAMETER C BY USING THE CROSS-VALIDATION METHOD
C     THROUGH THE IMSL MINIMIZING ROUTINE ZXLSF WHICH
C     CAN BE REPLACED BY ANOTHER MINIMIZING ROUTINE.
C     THE CALLING PROGRAM MUST INCLUDE THE STATEMENT
C     "CALL ZXLSF(F,X,STEP,BOUND,XACC,MAXFN,IER)".
C     DOUBLE PRECISION IS USED TO IMPROVE THE ACCURACY
C     OF ESTIMATION.
C
C
C             INPUT VARIABLES ARE:
C     Y     = ARRAY OF OBSERVED VALUES ( OREDERED IN TIME )
C     L     = ARRAY OF TIME-VALUES ( ORDERED IN TIME )
C     M     = DIM. OF ARRAYS Y AND L
C     NP    = THE ORDER OF THE LOCALLY POLYNOMIAL FIT PLUS 1;
C             IN THIS CASE, NP=1, THAT IS LOCALLY CONSTANT FIT.
C     NP2   = 2 x NP
C     NP1   = NP2 - 1
C     SC    = THE SCALING FACTOR
C     PROD  = WORKING ARRAY OF DIM. NP2
C     ATY   = WORKING ARRAY OF DIM. NP
C     ARRAYS ATA AND ATAI OF DIM. NP x NP, AND WORKING ARRAY
C     IPERM OF DIM. NP2 ARE USED IN THE SUBROUTINE INV
C     ( ALL THE ABOVE VARIABLES MUST BE DEFINED IN THE CALLING
C     PROGRAM )
C
C             OUTPUT VARIABLES ARE:
C     C     = ESTIMATE OF THE SMOOTHING PARAMETER
C     SUM   = CROSS-VALIDATED SUM OF SQUARED ERRORS
C
C             SUBROUTINES CALLED BY F ARE:
C     INV (CAN BE FOUND IN THE DOCUMENTATION: UBC MATRIX).
C     INV CAN BE REPLACED BY ANOTHER MATRIX INVERSE ROUTINE
C     WHERE ATA IS INPUT AND ATAI IS OUTPUT.
C
      REAL*8 ATA(1 ,1 ),ATAI(1 ,1 ),ATY(1 ),PROD(2 ),Y(80),
     *L(80),IPERM(2 )
C
C     THE DIMENSIONS OF THE ABOVE ARRAYS MUST BE SPECIFIED;
C     IN THIS CASE, NP=1, THAT IS LOCALLY CONSTANT FIT.
C
      REAL*8 SC,Z,C,SUM,TEMP,V,W,P,YY,BB
      COMMON Y,L,M,NP,SC,NP1,NP2
C
C     THE ABOVE "COMMON" STATEMENT IS USED IN ORDER TO PASS
C     THE PARAMETERS FROM THE CALLING ROUTINE TO THIS
C     ROUTINE. THE CALLING PROGRAM MUST INCLUDE THE SAME
C     "COMMON" STATEMENT.
C
      Z=DABS(C)
```

```
      SUM=0.D0
      DO 25 J=1,M
      BB=0.D0
      DO 5 JJ=1,NP
    5 ATY(JJ)=0.D0
      DO 10 II=1,NP1
   10 PROD(II)=0.D0
      NT=L(J)
      DO 20 I=1,M
      IF(I.EQ.J) GO TO 20
      P=SC*(L(I)-NT)
      W=1.D0/(1.D0+Z*(P**NP2))
      V=W
      DO 12 II=1,NP1
      PROD(II)=PROD(II)+V
      V=V*P
   12 CONTINUE
      V=W
      YY=Y(I)
      DO 14 JJ=1,NP
      ATY(JJ)=ATY(JJ)+V*YY
      V=V*P
   14 CONTINUE
   20 CONTINUE
      DO 22 II=1,NP
      DO 22 JJ=1,NP
      ATA(II,JJ)=PROD(II+JJ-1)
   22 CONTINUE
C
C     COMPUTE THE REGRESSION COEFFICIENTS
C
      CALL INV(NP,NP,ATA,IPERM,NP,ATAI,DET,JEXP,COND)
      DO 30 I=1,NP
      BB=BB+ATAI(1,I)*ATY(I)
C
C     BB IS THE CROSS-VALIDATORY ESTIMATE OF THE SMOOTH
C
   30 CONTINUE
      SUM=SUM+(Y(J)-BB)**2
   25 CONTINUE
      WRITE(6,99) Z,SUM
   99 FORMAT(E18.7,2X,E15.8)
C
C     Z IS THE ABSOLUTE VALUE OF C AND SUM IS THE CROSS-
C     VALIDATED SUM OF SQUARED ERRORS. THEY ARE OUTPUT SO
C     THAT THE VALUE OF C WHICH MINIMIZES SUM CAN BE
C     DETERMINED.
C
      F=SUM
      RETURN
      END
```

```
      SUBROUTINE BFIT(Y,L,B,ATA,ATY,PROD,ATAI,IPERM,TT,TTT
     *                CC,TA,M,NP,SC,C,NP2,N)
C
C     THIS IS A ROUTINE FOR COMPUTING THE SMOOTHING
C     PARAMETER C BY USING THE BACKFITTING METHOD.
C
C            INPUT VARIABLES ARE:
C     Y    = ARRAY OF OBSERVED VALUES (ORDERED IN TIME)
C     L    = ARRAY OF TIME-VALUES (ORDERED IN TIME)
C     M    = DIM. OF ARRAYS Y AND L
C     NP   = THE ORDER OF THE LOCALLY PLOYNOMIAL FIT PLUS 1;
C            IN THIS CASE, NP=11
C     SC   = THE SCALING FACTOR
C     C    = THE SMOOTHING PARAMETER OF THE 10TH ORDER LOCALLY
C            POLYNOMIAL FIT; IN THIS CASE, C=0.0D0
C     TA   = ARRAY OF THE DERIVATIVES OF THE SMOOTH OF
C            DIM. M AND N
C     NP2  = 2 x NP
C     N    = NP - 1
C     PROD = WORKING ARRAY OF DIM. NP2
C     B, TT, TTT ARE WORKING ARRAYS OF DIM. NP
C     ATA AND ATAI OF DIM. NP x NP, ATY OF DIM. NP AND
C     IPERM OF DIM. NP2 ARE ARRAYS USED IN THE SUBROUTINES
C     INV AND DGMATV
C     ( ALL THE ABOVE VARIABLES MUST BE DEFINED IN THE
C     CALLING PROGRAM. )
C
C            OUTPUT VARIABLES ARE:
C     CC(K)= SMOOTHING PARAMETER OF THE KTH ORDER LOCALLY
C            POLYNOMIAL FIT, K = 1,...,N
C
C            SUBROUTINES CALLED BY BFIT ARE:
C     INV AND DGMATV (CAN BE FOUND IN THE DOCUMENTATION:
C     UBC MATRIX). INV CAN BE REPLACED BY ANOTHER MATRIX
C     INVERSE ROUTINE WHERE ATA IS INPUT AND ATAI IS
C     OUTPUT. DGMATV CAN BE REPLACED BY ANOTHER MATRIX
C     MULTIPICATION ROUTINE WHERE ATAI AND ATY ARE INPUTS
C     AND B IS OUTPUT.
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION Y(M),L(M),B(NP),ATA(NP,NP),ATY(NP)
     *,PROD(NP2),ATAI(NP,NP),IPERM(NP2),TT(N),TTT(N),CC(N),
     *,TA(M,N)
      NP1=NP2-1
      DD=0.0D0
C
C     COMPUTE THE SAMPLE VARIANCE OF THE NOISE DD AND
C     THE DERIVATIVES OF THE SMOOTH TA
C
      DO 2 I=1,N
      TT(I)=0.0D0
 2    TTT(I)=0.0D0
```

```
      DO 25 J=1,M
      DO 5 JJ=1,NP
 5    ATY(JJ)=0.D0
      DO 10 II=1,NP1
 10   PROD(II)=0.D0
      NT=L(J)
      DO 19 I=1,M
      P=SC*(L(I)-NT)
      W=1.D0/(1.D0+C*(P**NP2))
      V=W
      DO 12 II=1,NP1
      PROD(II)=PROD(II)+V
      V=V*P
 12   CONTINUE
      V=W
      YY=Y(I)
      DO 14 JJ=1,NP
      ATY(JJ)=ATY(JJ)+V*YY
      V=V*P
 14   CONTINUE
 19   CONTINUE
      DO 22 II=1,NP
      DO 22 JJ=1,NP
      ATA(II,JJ)=PROD(II+JJ-1)
 22   CONTINUE
      CALL INV(NP,NP,ATA,IPERM,NP,ATAI,DET,JEXP,COND)
      CALL DGMATV(ATAI,ATY,B,NP,NP,NP)
      TEMP=ATAI(1,1)
      DD=DD+(Y(J)-B(1))**2/(1.D0+TEMP)
      DO 21 K=1,N
      TA(J,K)=B(K+1)
 21   TT(K)=TT(K)+TA(J,K)
 25   CONTINUE
      DD=DD/(M-NP)
C
C     COMPUTE THE SMOOTHING PARAMETERS CC(K), K = 1,...,N
C
      DO 44 I=1,N
 44   TT(I)=TT(I)/M
      DO 55 I=1,N
      DO 66 K=1,M
 66   TTT(I)=TTT(I)+(TA(K,I)-TT(I))**2
 55   CONTINUE
      DO 77 I=1,N
 77   TTT(I)=TTT(I)/(M-1)
      DO 88 K=1,N
      CC(K)=TTT(K)/(DD)
 88   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SMOOTH(Y,L,MM,M,NP,SC,C,KK,UL,VL,
     *PROD,ATY,TEM,A,ATA,ATAI,IPERM,NP1,NP2)
C
C     THIS IS THE ROUTINE FOR COMPUTING THE INTERPOLANTS
C     AND THE PREDICTANDS OF THE SMOOTH CURVE WITH THE
C     CORRESPONDING CREDIBILITY INTERVALS.
C
C           INPUT VARIABLES ARE:
C     Y    = ARRAY OF OBSERVED VALUES (ORDERED IN TIME)
C     L    = ARRAY OF TIME-VALUES (ORDERED IN TIME)
C     MM   = THE TOTAL NUMBER OF THE INTERPOLANTS AND THE
C            PREDICTANDS OF THE SMOOTH CURVE
C     M    = TOTAL NUMBER OF OBSERVATIONS OF Y
C     NP   = THE ORDER OF THE LOCALLY POLYNOMIAL FIT PLUS 1
C     SC   = THE SCALING FACTOR
C     C    = THE CROSS-VALIDATORY OR THE BACKFITTING
C            SMOOTHING PARAMETER
C     NP2  = 2 x NP
C     NP1  = NP2 - 1
C     KK   = THE NUMBER OF PREDICTANDS BEYOND THE EITHER
C            SIDE OF THE RANGE OF DATA
C     PROD= WORKING ARRAY OF DIM. NP2
C     ATY  = WORKING ARRAY OF DIM. NP1
C     TEM  = WORKING ARRAY OF DIM. MM
C     ARRAYS ATA AND ATAI OF DIM. NP AND NP AND WORKING
C     ARRAY IPERM OF DIM NP2 ARE USED IN THE SUBROUTINE INV
C     ( ALL THE ABOVE VARIABLES MUST BE DEFINED IN THE
C       CALLING PROGRAM. )
C
C           OUTPUT VARIABLES ARE:
C     A    = ARRAY OF DIM. MM OF THE INTERPOLANTS OR THE
C            PREDICTANDS OF THE SMOOTH CURVE
C     UL   = ARRAY OF DIM. MM OF THE UPPER LIMITS OF THE
C            CREDIBILITY INTERVALS
C     VL   = ARRAY OF DIM. MM OF THE LOWER LIMITS OF THE
C            CREDIBILITY INTERVALS
C     SE   = STANDARD ERROR OF THE INTERPOLANTS OR THE
C            PREDICTANDS
C
C           SUBROUTINES CALLED BY SMOOTH ARE:
C     INV (CAN BE FOUND IN THE DOCUMENTATION: UBC MATRIX).
C     INV CAN BE REPLACED BY ANOTHER MATRIX INVERSE
C     ROUTINE WHERE ATA IS INPUT AND ATAI IS OUTPUT.
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION Y(MM),L(MM),ATA(NP,NP),ATY(NP),PROD(NP2)
     *,ATAI(NP,NP),IPERM(NP2),A(MM),TEM(MM),UL(MM),VL(MM)
      N=NP-1
      K1=KK+1
      K2=KK+M
      DD=0.D0
      DO 2 I=1,KK
```

```
      L(I)=5*(I-1)+300
 2    L(I+92)=5*(I-1)+730
C
C     COMPUTE THE SAMPLE VARIANCE OF THE NOISE DD
C
      DO 25 J=K1,K2
      BB=0.D0
      DO 5 JJ=1,NP
 5    ATY(JJ)=0.D0
      DO 10 II=1,NP1
 10   PROD(II)=0.D0
      NT=L(J)
      DO 19 I=K1,K2
      P=SC*(L(I)-NT)
      W=1.D0/(1.D0+C*(P**NP2))
      V=W
      DO 12 II=1,NP1
      PROD(II)=PROD(II)+V
      V=V*P
 12   CONTINUE
      V=W
      YY=Y(I)
      DO 14 JJ=1,NP
      ATY(JJ)=ATY(JJ)+V*YY
      V=V*P
 14   CONTINUE
 19   CONTINUE
      DO 22 II=1,NP
      DO 22 JJ=1,NP
      ATA(II,JJ)=PROD(II+JJ-1)
 22   CONTINUE
      CALL INV(NP,NP,ATA,IPERM,NP,ATAI,DET,JEXP,COND)
      DO 30 I=1,NP
      BB=BB+ATAI(1,I)*ATY(I)
 30   CONTINUE
      TEMP=ATAI(1,1)
      DD=DD+(Y(J)-BB)**2/(1.D0+TEMP)
 25   CONTINUE
      DD=DD/(M-NP)
C
C     COMPUTE THE STANDARD ERROR SE
C
      DO 29 J=1,MM
      BB=0.D0
      DO 6 JJ=1,NP
 6    ATY(JJ)=0.D0
      DO 16 II=1,NP1
 16   PROD(II)=0.D0
      NT=L(J)
      DO 49 I=K1,K2
      P=SC*(L(I)-NT)
      W=1.D0/(1.D0+C*(P**NP2))
```

```
      V=W
      DO 27 II=1,NP1
      PROD(II)=PROD(II)+V
      V=V*P
 27   CONTINUE
      V=W
      YY=Y(I)
      DO 74 JJ=1,NP
      ATY(JJ)=ATY(JJ)+V*YY
      V=V*P
 74   CONTINUE
 49   CONTINUE
      DO 82 II=1,NP
      DO 82 JJ=1,NP
      ATA(II,JJ)=PROD(II+JJ-1)
 82   CONTINUE
      CALL INV(NP,NP,ATA,IPERM,NP,ATAI,DET,JEXP,COND)
      DO 39 I=1,NP
      BB=BB+ATAI(1,I)*ATY(I)
 39   CONTINUE
      TEM(J)=ATAI(1,1)
      A(J)=BB
 29   CONTINUE
      DO 101 I=1,MM
      SE=DSQRT(DD*TEM(I))
C
C     COMPUTE THE CREDIBILITY INTERVALS
C
      UL(I)=A(I)+1.96*SE
      VL(I)=A(I)-1.96*SE
 101  CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PSE(SC,N1,N2,C,M,N4,NP,NP1,NP2,N,Y,L,CC,
     *               ATA,ATY,ATAI,PROD,B,IPERM,TT,TTT,TA,VAR,
     *               R,AA)
C
C     THIS IS A ROUTINE FOR COMPUTING THE VALUES OF PSE
C      --- PREDICT ONE OBSERVATION AHEAD.
C
C              INPUT VARIABLES ARE:
C     Y      = ARRAY OF OBSERVED VALUES (ORDERED IN TIME)
C     L      = ARRAY OF TIME-VALUES (ORDERED IN TIME)
C     M      = DIM. OF THE ARRAYS Y AND L
C     NP     = THE ORDER OF THE LOCALLY POLYNOMIAL FIT PLUS 1
C     C      = THE SMOOTHING PARAMETER OF THE 10TH ORDER
C              LOCALLY POLYNOMIAL FIT; IN THIS CASE, C=0.0D0
C     SC     = THE SCALING FACTOR
C     N1,N2  = THE STARTING LIMITS OF THE SPAN OF OBSERVATIONS
C     N      = NP - 1
C     NP2    = 2 x NP
C     NP1    = NP2 - 1
C     N4     = M - N2
C     B      = ARRAY OF THE DERIVATIVES OF THE SMOOTH
C     CC     = ARRAY OF THE SMOOTHING PARAMETERS OF DIFFERENT
C              ORDER LOCALLY POLYNOMIAL FITS
C     AA     = ARRAY OF THE ESTIMATES OF FUTURE OBSERVATIONS
C     VAR    = WORKING ARRAY OF DIM. M
C     PROD   = WORKING ARRAY OF DIM. NP2
C     TA     = WORKING ARRAY OF DIM. N2 AND N
C     TT,TTT= WORKING ARRAY OF DIM. N
C     ARRAYS ATA AND ATAI OF DIM. NP x NP, ARRAY ATY OF DIM.
C     NP AND ARRAY IPERM OF DIM. NP2 ARE USED IN THE
C     ROUTINES INV AND DGMATV ( CAN BE FOUND IN THE
C     DOCUMENTATION: UBC MATRIX ). INV CAN BE REPLACED BY
C     ANOTHER MATRIX INVERSE ROUTINE WHERE ATA IS INPUT AND
C     ATAI IS OUTPUT. DGMATV CAN BE REPLACED BY ANOTHER
C     MATRIX MULTIPICATION ROUTINE WHERE ATAI AND ATY ARE
C     INPUTS AND B IS OUTPUT.
C     ( ALL THE ABOVE VARIABLES MUST BE DEFINED IN THE
C       CALLING PROGRAM )
C
C              OUTPUT VARIABLES ARE:
C     R(I)   = ESTIMATED VALUE OF PSE OF THE ITH ORDER LOCALLY
C              POLYNOMIAL FIT, I = 1,...,N
C
C              SUBROUTINES CALLED BY PSE ARE:
C     BACK AND PRED (ATTACHED TO THIS ROUTINE)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION Y(M),L(M),AA(N,N4),CC(N,N4),ATA(NP,NP)
     *,ATY(NP),ATAI(NP,NP),PROD(NP2),B(NP),IPERM(NP2),TT(N4),
     *TTT(N4),TA(N2,N),VAR(M),R(N)
C
C     COMPUTE THE BACKFITTING VALUES OF C AND THE SAMPLE
```

```
C      VARIANCE OF THE NOISE
C
       CALL BACK(SC,N1,N2,C,M,N4,NP,NP1,NP2,N,Y,L,CC,ATA,
      *ATY,ATAI,PROD,B,IPERM,TT,TTT,TA,VAR)
C
C      COMPUTE THE PREDICTIVE VALUES OF FUTURE OBSERVATIONS
C
       CALL PRED(SC,NP,NP1,NP2,N,M,N1,N2,N4,Y,L,CC,ATA,ATY,
      *ATAI,PROD,IPERM,AA)
C
C      COMPUTE THE ESTIMATED VALUES OF PSE
C
       DO 100 I=1,N
       R(I)=0.0
       DO 101 J=1,N4
       R(I)=R(I)+(Y(N2+J)-AA(I,J))**2
  101  CONTINUE
       R(I)=R(I)/N4
  100  CONTINUE
       RETURN
       END
C
C
       SUBROUTINE BACK(SC,N1,N2,C,M,N4,NP,NP1,NP2,N,Y,L,CC,
      *ATA,ATY,ATAI,PROD,B,IPERM,TT,TTT,TA,VAR)
       IMPLICIT REAL*8(A-H,O-Z)
       DIMENSION Y(M),L(M),B(NP),ATA(NP,NP),PROD(NP2),
      *ATAI(NP,NP),IPERM(NP2),TT(N),TTT(N),CC(N,N4),TA(N2,N),
      *ATY(NP),VAR(M)
       K1=N1
       K2=N2
       DO 18 NN=1,N4
       DD=0.D0
       DO 2 I=1,N
       DO 3 J=1,N2
       TA(J,I)=0.D0
    3  CONTINUE
       TT(I)=0.D0
       TTT(I)=0.D0
    2  CONTINUE
       DO 25 J=K1,K2
       DO 5 JJ=1,NP
    5  ATY(JJ)=0.D0
       DO 10 II=1,NP1
   10  PROD(II)=0.D0
       NT=L(J)
       DO 19 I=K1,K2
       P=SC*(L(I)-NT)
       W=1.D0/(1.D0+C*(P**NP2))
       V=W
       DO 12 II=1,NP1
       PROD(II)=PROD(II)+V
```

```
      V=V*P
   12 CONTINUE
      V=W
      YY=Y(I)
      DO 14 JJ=1,NP
      ATY(JJ)=ATY(JJ)+V*YY
      V=V*P
   14 CONTINUE
   19 CONTINUE
      DO 22 II=1,NP
      DO 22 JJ=1,NP
      ATA(II,JJ)=PROD(II+JJ-1)
   22 CONTINUE
      CALL INV(NP,NP,ATA,IPERM,NP,ATAI,DET,JEXP,COND)
      CALL DGMATV(ATAI,ATY,B,NP,NP,NP)
      TEMP=ATAI(1,1)
      DD=DD+(Y(J)-B(1))**2/(1.D0+TEMP)
      DO 21 K=1,N
      TA(J,K)=B(K+1)
   21 TT(K)=TT(K)+TA(J,K)
   25 CONTINUE
      DD=DD/(N2-NP)
      DO 44 I=1,N
   44 TT(I)=TT(I)/N2
      DO 55 I=1,N
      DO 66 K=K1,K2
   66 TTT(I)=TTT(I)+(TA(K,I)-TT(I))**2
   55 CONTINUE
      DO 47 I=1,N
   47 TTT(I)=TTT(I)/(N2-1)
      DO 88 K=1,N
      CC(K,NN)=TTT(K)/DD
   88 CONTINUE
      VAR(NN)=DD
      K1=K1+1
      K2=K2+1
   18 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE PRED(SC,NP,NP1,NP2,N,M,N1,N2,N4,Y,L,CC,ATA
     *,ATY,ATAI,PROD,IPERM,AA)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION Y(M),L(M),ATA(NP,NP),PROD(NP2),AA(N,N4)
     *,ATAI(NP,NP),IPERM(NP2),CC(N,N4),ATY(NP)
      K1=N1
      K2=N2
      DO 71 IP=1,N4
      DO 72 JP=1,N
      NQ=JP
      NQ2=2*JP
```

```
      NQ1=NQ2-1
      BB=0.D0
      DO 74 JJ=1,NQ
   74 ATY(JJ)=0.D0
      DO 75 II=1,NQ1
   75 PROD(II)=0.D0
      KT=L(N2+IP)
      DO 76 J=K1,K2
      P=SC*(L(J)-KT)
      W=1.D0/(1.D0+CC(JP,IP)*(P**NQ2))
      V=W
      DO 77 II=1,NQ1
      PROD(II)=PROD(II)+V
      V=V*P
   77 CONTINUE
      V=W
      YY=Y(J)
      DO 78 JJ=1,NQ
      ATY(JJ)=ATY(JJ)+V*YY
      V=V*P
   78 CONTINUE
   76 CONTINUE
      DO 79 II=1,NQ
      DO 79 JJ=1,NQ
      ATA(II,JJ)=PROD(II+JJ-1)
   79 CONTINUE
      CALL INV(NQ,NP,ATA,IPERM,NP,ATAI,DET,JEXP,COND)
      DO 80 I=1,NQ
      BB=BB+ATAI(1,I)*ATY(I)
   80 CONTINUE
      AA(JP,IP)=BB
   72 CONTINUE
      K1=K1+1
      K2=K2+1
   71 CONTINUE
      RETURN
      END
```