LOGSPLINE ESTIMATION OF A POSSIBLY MIXED SPECTRAL DISTRIBUTION

CHARLES KOOPERBERG
University of Washington

CHARLES J. STONE
University of California at Berkeley

YOUNG K. TRUONG
University of North Carolina at Chapel Hill

Revised: November 6, 1994

Abstract. Cubic splines and indicator functions are used to estimate the spectral density function and line spectrum, respectively, for a stationary time series. A fully automatic procedure involving maximum likelihood, stepwise addition and deletion of basis functions, and BIC is used to select the final model.

Keywords. Cubic splines, line spectrum, model selection, spectral density function, stationary time series.

1. INTRODUCTION

The problem of estimating the spectral distribution for a stationary time series is of fundamental importance in statistics. If this distribution is absolutely continuous, its density function can be estimated by a variety of methods, the most popular being window, AR and ARMA estimates. Window estimates are obtained by smoothing the periodogram using "window" functions, while AR and ARMA estimates are obtained by fitting parametric AR and ARMA models using "automatic" model selection procedures such as AIC and BIC; see Priestley (1981, Chapters 6 and 7).

It is known that the periodogram is not a consistent estimate of the spectral density function and that consistency can be achieved by smoothing the periodogram ordinates, the degree of smoothing being controlled by the window width. Larger window widths smooth out the noise, but also tend to distort the details of the signal, while smaller window widths tend to yield estimates with spurious features. Recent approaches to the window width selection problem involve the application of cross-validation (Beltrão and Bloomfield, 1987; Hurvich and Beltrão, 1990), the bootstrap (Swanepoel and van Wyk, 1986; Franke and Härdle, 1992; Politis and Romano, 1992) and regression methods (Wahba, 1980; Pawitan and Gangopadhyay, 1991).

In the AR approach the estimated spectral density function has the parametric form of the spectrum of an autoregressive process, and the resulting fit is better when the spectral density function can accurately be approximated by such a form. However, this procedure may yield poor estimates when it is used to fit simple MA models; see Beamish and Priestley (1981) and Figure 3 below. The ARMA approach extends the AR approach by approximating the spectral density function with the spectrum of an ARMA model. Since many more spectral density functions can accurately be approximated by ARMA models, not surprisingly, ARMA estimates are often better than AR estimates in fitting non-AR models. However, in our experience, numerical procedures for determining maximum likelihood estimates of ARMA parameters are typically much less stable and far more computer intensive than the other approaches discussed in this paper, making the ARMA approach considerably less attractive. See Section 7 for more details.

If the spectral distribution is possibly mixed, there are two general approaches to its estimation that have previously been discussed. One approach is to apply a method that was actually designed for the absolutely continuous case and hope that the estimate has sharp peaks centered near the atoms. The AR methods are generally used for this purpose since they typically yield sharper peaks corresponding to atoms than window estimates, while they are numerically more stable than ARMA estimates. Mackisack and Poskitt (1990) give an asymptotic justification for the AR procedure. In the engineering literature there have been various proposals to make AR methods more sensitive to atoms in the spectral distribution (see, for example, Stoica, Moses, Söderström and Li, 1991).

The alternative approach is first to test whether atoms are present. If so, their locations and masses are estimated, the corresponding components are filtered out, and the spectral

density function is estimated from the filtered time series. Two early references on this approach are Priestley (1962a) and Priestley (1962b). See Priestley (1981; Chapter 8) for an overview.

In this paper, we propose a new automatic procedure for estimating a possibly mixed spectral distribution. This method has the advantage of AR and ARMA estimates in being automatic and the advantage of the test-based procedures in giving explicit estimates for the masses and locations of the atoms. In the procedure to be studied here, the logarithm of the spectral density function is modeled as a polynomial spline, the unknown parameters of which are estimated by maximizing an approximation of the log-likelihood function. The performance of our procedure is compared to that of AR and ARMA estimates using both simulated and real time series data.

2. MIXED SPECTRA

Consider a real-valued second order stationary time series X_t with mean $E(X_t) = E(X_0)$ and covariance function $\gamma(u) = \mathrm{cov}(X_t, X_{t+u})$. Assume that the time series has the form

$$X_t = \sum_{j=1}^{p} R_j \cos(t\lambda_j + \phi_j) + Y_t. \tag{1}$$

Here $0 < \lambda_j \leq \pi$; ϕ_j are independent and uniformly distributed on $[-\pi, \pi]$; R_j are independent, non-negative random variables such that R_j^2 has positive mean $4\rho_j$; and Y_t is a second-order stationary time series such that $E(Y_t) = E(X_0)$ and

$$\sum_u |\gamma_c(u)| < \infty, \tag{2}$$

where $\gamma_c(u) = \mathrm{cov}(Y_t, Y_{t+u})$.

For a time series X_t that satisfies these conditions, we now first define several functions related to the spectral distribution. In particular, the *spectral density function* of the time series is given by

$$f_c(\lambda) = \frac{1}{2\pi} \sum_{u=-\infty}^{\infty} \gamma_c(u) \exp(iu\lambda), \qquad -\pi \leq \lambda \leq \pi, \tag{3}$$

which can be extended to $(-\infty, \infty)$ in the obvious manner so as to be periodic with period 2π; its *line spectrum* is given by

$$f_d(\lambda) = \begin{cases} \rho_j & \text{if } \lambda = \pm\lambda_j, \\ 0 & \text{otherwise;} \end{cases}$$

and its *spectral distribution function* is given by

$$F(\lambda) = \int_{-\pi}^{\lambda} f_c(\omega) d\omega + \sum_{\omega \leq \lambda} f_d(\omega), \qquad -\pi \leq \lambda \leq \pi.$$

The autocovariance is given in terms of the spectral distribution function by

$$\gamma(u) = \int_{-\pi}^{\pi} \exp(iu\lambda) dF(\lambda).$$

Note that f_c and f_d are symmetric about zero. If $p = 0$, then $\sum_u |\gamma(u)| < \infty$, $f_d = 0$, and the spectral distribution is absolutely continuous. We refer to $\pm\lambda_j$, $1 \le j \le p$, as the *atoms* of the spectral distribution and to ρ_j as the *mass* of the distribution at $\pm\lambda_j$. Note that if R_j equals $2\sqrt{\rho_j}$ in (1), then the time series is the mixed model discussed in Mackisack and Poskitt (1990).

The objective of this paper is to develop an adaptive methodology for estimating the spectral distribution for the series X_t. In particular, we will estimate the log of the spectral density function with cubic splines and the line spectrum by a sum of Dirac delta functions. The estimation procedure to be described in Section 4 will be based upon *the periodogram*

$$I^{(T)}(\lambda) = (2\pi T)^{-1} \left| \sum_{t=0}^{T-1} \exp(-i\lambda t) X_t \right|^2, \qquad -\pi \le \lambda \le \pi,$$

corresponding to the realization $X_0, \ldots, X_{T-1}$ of the time series. If F is absolutely continuous, then the periodogram can be regarded as an (inconsistent) estimate of the spectral density function. In particular, if $\sum_u |u|\,|\gamma(u)| < \infty$ then

$$E[I^{(T)}(\lambda)] = f_c(\lambda) + (2\pi T)^{-1} \left[\frac{\sin T\lambda/2}{\sin \lambda/2} \right]^2 E(X_0)^2 + O(T^{-1}),$$

where the $O(T^{-1})$ is uniform in λ (Brillinger, 1981).

It is convenient to refer to the function $f = f_c + \frac{T}{2\pi} f_d$ as the *mean function*. Under the assumptions that the atoms are all of the form $2\pi j/T$ for some integer j and that the time series is Gaussian, it can be shown that

$$I^{(T)}\left(\frac{2\pi j}{T}\right) = \left[f_c\left(\frac{2\pi j}{T}\right) + \frac{T}{2\pi} f_d\left(\frac{2\pi j}{T}\right) \right] W_j = f\left(\frac{2\pi j}{T}\right) W_j, \qquad 1 \le j \le T/2,$$

where W_j has approximately the exponential distribution with mean one if $j < T/2$ and approximately the χ^2 distribution with one degree of freedom if T is even and $j = T/2$, and W_j, $1 \le j \le T/2$, are asymptotically independent; see Brillinger (1981, Theorem 5.2.6).

Since f_c and f_d are symmetric about zero, from now on we limit our attention to the interval $[0, \pi]$. Observe that if the indicated derivatives of f_c exist, then $f_c'(0)$, $f_c'''(0)$, $f_c'(\pi)$ and $f_c'''(\pi)$ all equal zero. Let $\delta_a(\lambda)$ equal one or zero according as $\lambda = a$ or $\lambda \ne a$. Set $\varphi = \log f$ and $\varphi_c = \log f_c$. Then $\varphi = \varphi_c + \varphi_d$, where $\varphi_d = \theta_1 \delta_{\lambda_1} + \cdots + \theta_p \delta_{\lambda_p}$ with $\theta_1, \ldots, \theta_p > 0$. Moreover, $f_d = (2\pi/T) f_c [\exp(\varphi_d) - 1]$. In the next section we will use cubic splines to obtain a finite-dimensional approximation to φ_c and hence to φ.

3. LINEAR MODELS

First we describe the space of splines that will be used to model the log of the spectral density function. Given the positive integer K_c and the sequence $t_1, \ldots, t_{K_c}$ of knots with $0 \le t_1 < \cdots < t_{K_c} \le \pi$, let G_c be the K_c-dimensional space of twice continuously differentiable functions s on $[0, \pi]$ such that the restriction of s to each of the intervals $[0, t_1], [t_1, t_2], \ldots, [t_{K_c-1}, t_{K_c}], [t_{K_c}, \pi]$ is a cubic polynomial, the first derivative of s is zero at 0 and π, the third derivative of s is zero at 0 unless $t_1 = 0$, and the third derivative of s is zero at π unless $t_{K_c} = \pi$. (In particular, if $K_c = 1$, then G_c is the space of constant functions.) Note that the functions in G_c can be extended to splines on $(-\infty, \infty)$ that are symmetric about zero, periodic with period 2π, have a knot at zero if and only if $t_1 = 0$, and have a knot at π if and only if $t_{K_c} = \pi$. Let $B_1, \ldots, B_{K_c}$ be a basis of G_c. In our implementation each of these basis functions is a linear combination of λ, λ^2 and $(\lambda - t_i)_+^3$, for $i = 1, \ldots, K_c$. Motivated by the properties of B-splines (de Boor, 1978) we choose these linear combinations such that the basis functions are nonzero on a bounded interval and that each knot is only used in at most five basis functions. Because of the restrictions on derivatives at 0 and π there are a few more restrictions on the basis functions.

Next, we describe the space that will be used indirectly to model the line spectrum. Given the nonnegative integer K_d and the increasing sequence $a_1, \ldots, a_{K_d}$ of members of $\{2\pi j/T : 1 \le j \le T/2\}$, let G_d be the K_d-dimensional space of nonnegative functions s on $[0, \pi]$ such that $s = 0$ except at $a_1, \ldots, a_{K_d}$. Set $B_{j+K_c}(\lambda) = \delta_{a_j}(\lambda)$ for $1 \le j \le K_d$. Then $B_{K_c+1}, \ldots, B_K$ form a basis of G_d, where $K = K_c + K_d$.

Let G be the space spanned by $B_1, \ldots, B_K$. Set

$$\varphi_c(\cdot; \boldsymbol{\beta}_c) = \beta_1 B_1 + \cdots + \beta_{K_c} B_{K_c}$$

for $\boldsymbol{\beta}_c = (\beta_1, \ldots, \beta_{K_c})^t \in \mathbb{R}^{K_c}$,

$$\varphi_d(\cdot; \boldsymbol{\beta}_d) = \beta_{K_c+1} B_{K_c+1} + \cdots + \beta_K B_K$$

for $\boldsymbol{\beta}_d = (\beta_{K_c+1}, \ldots, \beta_K)^t$ with $\beta_{K_c+1}, \ldots, \beta_K \ge 0$, and

$$\varphi(\cdot; \boldsymbol{\beta}) = \varphi_c(\cdot; \boldsymbol{\beta}_c) + \varphi_d(\cdot; \boldsymbol{\beta}_d)$$

for $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_K)^t$. We use $\varphi_c(\cdot; \boldsymbol{\beta}_c)$ to model the logarithm of the spectral density function and $\varphi(\cdot; \boldsymbol{\beta})$ to model the logarithm of the mean function. Thus, the spectral density function corresponding to $\boldsymbol{\beta}$ is given by $f_c(\cdot; \boldsymbol{\beta}_c) = \exp \varphi_c(\cdot; \boldsymbol{\beta}_c)$, the mean function is given by $f(\cdot; \boldsymbol{\beta}) = \exp \varphi(\cdot; \boldsymbol{\beta})$, and the line spectrum is given by

$$f_d(\cdot; \boldsymbol{\beta}_c) = \frac{2\pi}{T} f_c(\cdot; \boldsymbol{\beta}_c)[\exp(\varphi_d(\cdot; \boldsymbol{\beta}_d)) - 1].$$

Let $Y = f(\lambda; \boldsymbol{\beta})W$, where W has the exponential distribution with mean one when $0 < \lambda < \pi$ and it has the χ^2 distribution with one degree of freedom when $\lambda = \pi$. The

log-likelihood corresponding to the observed value y of Y is given by

$$\psi(y,\lambda;\boldsymbol{\beta}) = \left(\frac{\delta_\pi(\lambda)}{2} - 1\right)\left[\sum_{j=1}^{K}\beta_j B_j(\lambda) + y\exp\left(-\sum_{j=1}^{K}\beta_j B_j(\lambda)\right)\right]$$

for $0 < \lambda \leq \pi$ and $y \geq 0$, where we have ignored a term that does not depend on $\boldsymbol{\beta}$. Observe that

$$\frac{\partial}{\partial\beta_k}\psi(y,\lambda;\boldsymbol{\beta}) = \left(\frac{\delta_\pi(\lambda)}{2} - 1\right)B_k(\lambda)\left[1 - y\exp\left(-\sum_{j=1}^{K}\beta_j B_j(\lambda)\right)\right]$$

for $1 \leq k \leq K$, $0 < \lambda \leq \pi$ and $y \geq 0$. Observe also that

$$\frac{\partial^2}{\partial\beta_k\partial\beta_l}\psi(y,\lambda;\boldsymbol{\beta}) = \left(\frac{\delta_\pi(\lambda)}{2} - 1\right)yB_k(\lambda)B_l(\lambda)\exp\left(-\sum_{j=1}^{K}\beta_j B_j(\lambda)\right)$$

for $1 \leq k, l \leq K$, $0 < \lambda \leq \pi$ and $y \geq 0$. It follows from the last result that $\psi(y,\lambda;\cdot)$ is a concave function for $y \geq 0$ and $0 < \lambda \leq \pi$.

4. ESTIMATION

Let $X_0, \ldots, X_{T-1}$ be a realization of length T of the time series. For $1 \leq j \leq T/2$, let

$$I_j = (2\pi T)^{-1}\left|\sum_{t=0}^{T-1}\exp(-i2\pi jt/T)X_t\right|^2$$

be the value of the corresponding periodogram at the (angular) frequency $\lambda = 2\pi j/T$. The (approximate) log-likelihood function corresponding to the periodogram and the K-parameter model for the logarithm of the mean function is given by

$$\ell(\boldsymbol{\beta}) = \sum_j \psi\left(I_j, \frac{2\pi j}{T}; \boldsymbol{\beta}\right), \qquad \boldsymbol{\beta} \in I\!R^K \text{ with } \beta_{K_c+1}, \ldots, \beta_K \geq 0. \tag{4}$$

Moreover,

$$\frac{\partial}{\partial\beta_k}\ell(\boldsymbol{\beta}) = \sum_j \frac{\partial}{\partial\beta_k}\psi\left(I_j, \frac{2\pi j}{T}; \boldsymbol{\beta}\right)$$

for $1 \leq k \leq K$ and $\boldsymbol{\beta} \in I\!R^K$ with $\beta_{K_c+1}, \ldots, \beta_K \geq 0$. Also,

$$\frac{\partial^2}{\partial\beta_k\partial\beta_l}\ell(\boldsymbol{\beta}) = \sum_j \frac{\partial^2}{\partial\beta_k\partial\beta_l}\psi\left(I_j, \frac{2\pi j}{T}; \boldsymbol{\beta}\right)$$

for $1 \leq k, l \leq K$ and $\boldsymbol{\beta} \in I\!R^K$ with $\beta_{K_c+1}, \ldots, \beta_K \geq 0$. Note that (4) is equivalent to the Whittle likelihood (Whittle, 1961). The maximum likelihood estimate $\widehat{\boldsymbol{\beta}} = (\hat{\beta}_1, \ldots, \hat{\beta}_K)^T$ is given as usual by

$$\ell(\widehat{\boldsymbol{\beta}}) = \max_{\boldsymbol{\beta}\in I\!R^K}\ell(\boldsymbol{\beta}),$$

the log-likelihood of the model is given by $\hat{\ell} = \ell(\widehat{\boldsymbol{\beta}})$, and the maximum likelihood estimate of the mean function is given by $\hat{f}(\lambda) = f(\lambda; \widehat{\boldsymbol{\beta}})$. Similarly, the maximum likelihood estimates of the spectral density function and line spectrum are given by $\hat{f}_c(\cdot) = f_c(\cdot; \widehat{\boldsymbol{\beta}}_{\boldsymbol{c}})$ and $\hat{f}_d(\cdot) = f_d(\cdot, \widehat{\boldsymbol{\beta}}_{\boldsymbol{d}})$, where $\widehat{\boldsymbol{\beta}}_{\boldsymbol{c}} = (\hat{\beta}_1, \ldots, \hat{\beta}_{K_c})^T$ and $\widehat{\boldsymbol{\beta}}_{\boldsymbol{d}} = (\hat{\beta}_{K_c+1}, \ldots, \hat{\beta}_K)^T$.

Let $\mathbf{S}(\boldsymbol{\beta})$ denote the score at $\boldsymbol{\beta}$; that is, the K-dimensional column vector with entries $\partial\ell(\boldsymbol{\beta})/\partial\beta_j$. Let $\mathbf{H}(\boldsymbol{\beta})$ denote the Hessian at $\boldsymbol{\beta}$; that is, the $K \times K$ matrix with entries $\partial^2\ell(\boldsymbol{\beta})/\partial\beta_j\partial\beta_k$. The Newton-Raphson method for computing $\widehat{\boldsymbol{\beta}}$ is to start with $\widehat{\boldsymbol{\beta}}^{(\mathbf{0})}$ and iteratively determine $\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m+1})}$ from $\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})}$ according to the formula

$$\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m+1})} = \widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})} - [\mathbf{H}(\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})})]^{-1}\mathbf{S}(\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})}).$$

Here we employ the Newton-Raphson method with step-halving, in which $\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m+1})}$ is determined from $\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})}$ according to the formula

$$\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m+1})} = \widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})} - 2^{-\nu}[\mathbf{H}(\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})})]^{-1}\mathbf{S}(\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})}),$$

where ν is the smallest nonnegative integer such that

$$\ell\left(\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})} - 2^{-\nu}[\mathbf{H}(\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})})]^{-1}\mathbf{S}(\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})})\right) \geq \ell\left(\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})} - 2^{-\nu-1}[\mathbf{H}(\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})})]^{-1}\mathbf{S}(\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})})\right).$$

Also, we let $\widehat{\boldsymbol{\beta}}^{(\mathbf{0})}$ be the least squares solution to the linear regression problem

$$\log I_j = \sum_k \beta_k B_k(2\pi j/T) + \epsilon_j, \qquad 1 \leq j \leq T/2.$$

We stop the iterations when $\ell(\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m+1})}) - \ell(\widehat{\boldsymbol{\beta}}^{(\boldsymbol{m})}) \leq 10^{-6}$.

5. SELECTION OF BASIS FUNCTIONS

Initially, we fit the constant model discussed in Section 3. The location of the one knot in this model is of no importance. For simplicity we take $t_1 = 0$. After the constant model is fit, we add one knot according to the procedure described below. Then we remove the knot at zero and are left with a constant model having one knot.

We continue by successively adding a knot or atom at each step. When searching for the location of the knot or atom in adding a basis function, we compute the Rao statistic for the addition of a knot, as described below, for every frequency $\lambda = 2\pi j/T$, $1 \leq j \leq T/2$, such that $|\lambda - t_k| \geq 2(2\pi/T)$ for all knots t_k already in the model, and we compute the Rao statistic for the addition of an atom for every frequency $\lambda = 2\pi j/T$, $1 \leq j \leq T/2$, such that $\lambda \neq a_k$ for all atoms a_k already in the model. To facilitate the computation of the Rao statistic, we add one extra basis function to our existing basis so that the K basis functions span exactly the right K dimensional space G and only one of the basis functions depends on the new knot or atom. To prevent atoms with extremely small mass from entering the

model, some locations are ruled ineligible for atoms; see Section 10.6 for the details. We add the knot or eligible atom that has the largest Rao statistic, after which we redefine the basis functions of G.

Given a model with $K-1$ basis functions and a potential new knot or atom, let $\widehat{\boldsymbol{\beta}}_0$ be the maximum likelihood estimate of the K-dimensional coefficient vector subject to the constraint that the coefficient of the new basis function equals zero. Then the corresponding Rao statistic ((6e.3.6) of Rao, 1973) equals $[\mathbf{S}(\widehat{\boldsymbol{\beta}}_0)]^t[\mathbf{I}(\widehat{\boldsymbol{\beta}}_0)]^{-1}\mathbf{S}(\widehat{\boldsymbol{\beta}}_0)$, where $\mathbf{I}(\widehat{\boldsymbol{\beta}}_0) = -\mathbf{H}(\widehat{\boldsymbol{\beta}}_0)$ with $\mathbf{S}(\cdot)$ and $\mathbf{H}(\cdot)$ corresponding to the model with K basis functions. Issues involved in the efficient simultaneous computation of such Rao statistics are discussed in Section 10.1. Then, in Section 10.2, the Rao statistic is derived as the increase in the quadratic approximation to the log-likelihood function that would result from adding the knot or atom under consideration.

Example. Let $T = 200$. Assume that at sometime during the stepwise addition stage of the algorithm the LSPEC model has knots at frequencies $2\pi 9/200$, $2\pi 16/200$ and $2\pi 45/200$ and an atom at frequency $2\pi 30/200$. Then LSPEC will compute the Rao statistic for adding a knot at the frequencies $2\pi j/200$, $j = 1, \ldots, 6, 12, 13, 19, \ldots, 42, 48, \ldots, 100$. The other frequencies of the form $2\pi j/200$ are ineligible either because there is already a knot at that frequency, or because they are within $2\pi 2/200$ of a knot. LSPEC will also compute the Rao statistic for adding an atom at the frequencies $2\pi j/200$, for $j = 1, \ldots, 29, 31, \ldots, 100$. Except for the present location of the atom, each location of the form $2\pi j/200$ is eligible.

Upon stopping the stepwise addition process (according to a rule that is described in Section 10.3), we proceed to stepwise deletion. Here we successively remove the least statistically significant among the K remaining knots and atoms until only one basis function remains. Note that since that the subspace based on $K-1$ knots and atoms is a proper subspace of dimension one smaller than the space using K knots, removing a knot is equivalent to putting down a restriction $\mathbf{c}^t\widehat{\boldsymbol{\beta}} = 0$, where $\mathbf{c}$ is a coefficient vector (similar to a vector of contrasts in hypothesis testing) determined by our choice of basis functions, discussed in Section 3. The statistical significance of a remaining knot or atom is now measured by the absolute value of its Wald statistic $\hat{\tau}/\mathrm{SE}(\hat{\tau})$, where $\hat{\tau} = \mathbf{c}^t\widehat{\boldsymbol{\beta}}$ and $\mathrm{SE}(\hat{\tau}) = \sqrt{\mathbf{c}^t[\mathbf{I}(\widehat{\boldsymbol{\beta}})]^{-1}\mathbf{c}}$; here $\mathbf{I}(\widehat{\boldsymbol{\beta}}) = -\mathbf{H}(\widehat{\boldsymbol{\beta}})$ with $\mathbf{H}(\cdot)$ corresponding to the model with K basis functions. In the context of the removal of an atom, $\hat{\tau}$ is the coefficient $\hat{\beta}$ of the corresponding basis function; in the context of the removal of a knot, $\hat{\tau}$ is the jump of the third derivative of $\sum_j \hat{\beta}_j B_j$ at the knot, where the B_j's are defined in terms of the K remaining knots. In Section 10.2, the square of the Wald statistic is derived as the decrease in the quadratic approximation to the log-likelihood function that would result from deleting the knot or atom under consideration.

Example (cont). If the same model as above was obtained during the stepwise deletion stage of the algorithm, LSPEC would compute the Wald statistic for the removal of each of the three knots as well as the Wald statistic for the removal of the atom. Depending on which Wald statistic is the smallest in absolute value, either one of the three knots or the atom could be removed.

During the combination of stepwise addition followed by stepwise deletion, we get a sequence of models indexed by ν with the νth model having p_ν parameters. Let $\hat{\ell}_\nu$ denote the log-likelihood of the νth model, and let $\mathrm{AIC}_{\alpha,\nu} = -2\hat{\ell}_\nu + \alpha p_\nu$ be the Akaike Information Criterion with penalty parameter α for this model. We select the model corresponding to the value $\hat{\nu}$ of ν that minimizes $\mathrm{AIC}_{\alpha,\nu}$ as the optimal model for cycle one. (Typically we do not allow atoms to enter the model during the first cycle.) Then we proceed with another cycle of stepwise addition followed by stepwise deletion, where the initial knots and atoms are those that were in the optimal model for cycle one. Thus we obtain an optimal model for cycle two. Since the initial knots of the second cycle could be removed during stepwise deletion, some of the knots and atoms that were found optimal during cycle one may no longer be in the model at the end of cycle two. We continue these cycles of a combination of stepwise addition followed by stepwise deletion until either the optimal model does not change or we reach a maximum number of cycles. In our experience, when the true underlying spectral distribution is mixed, one cycle of stepwise addition and deletion might not be enough to find the proper balance between the continuous component and the discrete component of the estimated spectral distribution. We have found in such situations that a two-cycle or multi-cycle procedure is more likely to find a good model.

Example (cont). Suppose that the same model above is the optimal model at the end of the second cycle of stepwise addition and stepwise deletion. If it was fit during the addition stage of the second cycle, there is no need to continue adding, since the algorithm was already in the same situation during the addition stage of the second cycle. However, if the model was fit during the deletion stage, it is possible that the fit would improve during another addition and deletion cycle. In particular, LSPEC would now examine the knots and atoms for addition that were mentioned above.

In light of Kooperberg and Stone (1992) and our experience in the present investigation, we recommend choosing $\alpha = \log(n)$ as in the Bayesian information criterion (BIC), where $n = [T/2]$ is the number of ordinates of the periodogram that are included in the log-likelihood.

6. USER INTERFACE

A program for implementing the logspline estimation of a spectral distribution (LSPEC) as described in this paper has been written in C and an interface based on S (see Becker, Chambers and Wilks, 1988; Chambers and Hastie, 1992) has also been developed[1]. The interface consists of seven S functions: `clspec`, `dlspec`, `lspec.fit`, `lspec.summary`, `lspec.plot`, `plspec` and `rlspec`. The function `lspec.fit` performs the model fitting and model selection tasks and supplies the modest output that is used as input to the other functions. The function `dlspec` gives the spectral density function and line spectrum corresponding to the

[1]LSPEC software is available from statlib. Send an email with the body `send lspec from S` to statlib@stat.cmu.edu

fit obtained by `lspec.fit`. Similarly, `plspec` gives the spectral distribution function corresponding to this fit, `clspec` gives the autocovariances and autocorrelations corresponding to the fit, and `rlspec` gives a realization of a stationary Gaussian time-series corresponding to the fitted spectral distribution. The function `lspec.summary` uses the output of `lspec.fit` to provide summary information about the fit. Finally, `lspec.plot` uses the output of `lspec.fit` directly to produce a plot of the spectral density function, line spectrum or spectral distribution function.

It should be noted that an approximate, nonadaptive version of logspline spectral density estimation (without atoms) is readily available in standard S. Specifically, set `freq` $= 2\pi j/T$ for $1 \le j \le T/2$, and `period` $= I_j$ for $1 \le j \le T/2$. Then the S command

```
fixed <- glm(period ~ bs(freq,k), family = quasi(log, "mu^2"))
```

fits a B-spline with k degrees of freedom to the spectral density function. However, the location and number of knots is not optimized, no atoms are included, and the first and third derivatives of the spectral density estimate at 0 and π are not constrained to equal zero.

7. SIMULATED EXAMPLES

In this section, our procedure is compared to AR and ARMA spectral estimates using the automatic model selection rules AIC and BIC. The comparison involves the ARMA model with orders p and q defined by $Y_t + \alpha_1 Y_{t-1} + \cdots + \alpha_p Y_{t-p} = \epsilon_t + \beta_1 \epsilon_{t-1} + \cdots + \beta_q \epsilon_{t-q}$, where $\epsilon_t \sim_{\text{iid}} N(0, \sigma^2)$. The spectral density function for this model is given by

$$f(\lambda) = \frac{\sigma^2}{2\pi} \frac{|1 + \beta_1 e^{-i\lambda} + \cdots + \beta_q e^{-iq\lambda}|^2}{|1 + \alpha_1 e^{-i\lambda} + \cdots + \alpha_p e^{-ip\lambda}|^2}.$$

The pth order AR estimate of the spectral density function is defined by

$$\hat{f}(\lambda) = \frac{\hat{\sigma}_p^2}{2\pi} |1 + \hat{\alpha}_1 e^{-i\lambda} + \cdots + \hat{\alpha}_p e^{-ip\lambda}|^{-2},$$

where $\hat{\alpha}_1, \ldots, \hat{\alpha}_p$ and $\hat{\sigma}_p^2$ are the Yule-Walker estimates of $\alpha_1, \ldots, \alpha_p$ and σ_p^2, respectively. In the ARAIC estimate the order p is chosen to minimize the Akaike information criterion $\text{AIC}(p) = n \log(\hat{\sigma}_p^2) + 2p$; in the ARBIC estimate the order p is chosen to minimize the (modified) Bayes information criterion $\text{BIC}(p) = n \log(\hat{\sigma}_p^2) + \log(n)p$; see Priestley (1981, Section 5.4). The ARAIC estimate of the spectral density function tends to be considerably more irregular than the LSPEC and ARBIC estimates when the true spectral distribution is absolutely continuous. Thus, in the interest of simplicity, we do not show ARAIC estimates in Figures 1–8 below. Note that in our examples, we computed AIC(p) and BIC(p) for $p = 0, 1, \ldots, 20$.

ARMA estimates for the spectrum can be defined in a similar fashion as the AR estimates above. While the Yule-Walker algorithm for AR estimates is stable, fast and provides a unique solution, algorithms to fit ARMA models based on maximum likelihood are less stable and very sensitive to starting values. In particular, both ARMA algorithms that we used, the one in S-Plus based on the algorithm from Ansley (1979) and the one available from TIMESLAB (Newton, 1988), iteratively optimize the likelihood function. Since the likelihood function may have more than one maximum, starting values are of crucial importance.

In our computations, when we wanted to fit an ARMA$(p, 0)$ model using maximum likelihood, we used the Yule-Walker estimates as starting values, while if we wanted to fit an ARMA(p, q) model with $q > 0$, we used as starting values the estimated parameters from a previously fitted ARMA$(p, q-1)$ model with an additional 0 added for the extra parameter. Nevertheless, we had serious problems obtaining ARMA estimates in a systematic fashion. We list some of the problems that we encountered using the TIMESLAB software (our experience in using the S-Plus software was very similar):

- In all our experiments (many more than discussed in this article) we tried to fit ARMA(p, q) models with $p = 0, 1, \ldots, 10$ and $q = 0, 1, \ldots, 5$. We set the maximum number of iterations to 5000 (see below for the effect of this on the CPU time). We used the starting values as described above. Nevertheless, 23% of the time the algorithm did not convergence, although we could use parameter estimates for the model selection procedure. More seriously, 2% of the time the algorithm returned an error state, so we were not able to use parameter values. (These numbers do not include the simulation study reported in Table 4, for which the percentages were much higher.) These problems are likely due to occasional factorization infeasibility. See Newton (1988, p.224) and Pawitan and O'Sullivan (1994). (If the maximum number of iterations were reduced to 1000 the algorithm would have converged only 44% of the time.)

  In our examples below we used the model with the smallest BIC or AIC value, irrespective of the model being obtained after convergence or when the maximum number of iterations was reached.

- We had three types of computer workstations available for our computations: Sun SPARC, SGI Indy and DEC Alpha. When we ran the identical code using the same data on different machines we typically got different results. Usually these differences come from models that had converged on one machine but not on the other; occasionally, however, two versions of the same program using virtually identical starting values claimed convergence while providing entirely different parameter estimates.

  In our examples below we used the results from the DEC Alpha computers, which provided the highest percentage of convergence.

- The choice of 5000 as the maximum number of iterations in the ARMA fit had a serious effect on the CPU time. For example, to fit all ARMA models using the TIMESLAB software for the example with $n = 500$ in Figure 1 below the program used 806.3

seconds CPU time on a Sun SPARCstation 2. Were the maximum iterations reduced to 1000 (more than doubling the number of cases with no convergence), the CPU time would be reduced to 197.9 seconds. By comparison, on the same machine, the AR procedure took 0.5 seconds CPU time, and the LSPEC procedure took 6.5 seconds CPU time. (If LSPEC were run with the options `updown=1` and `maxatoms=0` to allow for only one cycle of model fitting and no atoms, the CPU time would be reduced to 2.4 seconds.) This comparison is actually tainted *against* the AR and LSPEC procedures, since these procedures were run from within S-Plus, causing substantial overhead, while the ARMA procedure was run as a Fortran program.

We believe that because of these problems associated with ARMA spectrum estimation, the practical utility of the existing implementations of this estimation procedure is extremely limited. Nevertheless, in our examples below we will be comparing LSPEC estimates to ARBIC and ARMABIC estimates.

For Figure 1 we generated sequences of length 100 (left side) and 500 (right side) from the AR(3) process defined by $Y_t - 1.5Y_{t-1} + 0.7Y_{t-2} - 0.1Y_{t-3} = \epsilon_t$, where $\epsilon_t \sim_{\text{iid}} N(0,1)$. In this and later figures, the LSPEC estimates of the spectral density function are shown as solid lines, the ARBIC estimates are shown as short dashed lines, the ARMABIC estimates are shown as long dashed lines and the true spectral density functions are shown as dotted lines. Some relevant information about the estimates in Figures 1–4 is summarized in Table 1 below.

Observe that the ARBIC, ARMABIC and LSPEC estimates are very similar. This is typical for low order AR models, where the difference between the estimates is generally much smaller than the difference between either estimate and the true spectral density function. The main difference between the estimates is that the ARMABIC estimate for $T = 100$ is considerably higher for frequencies close to zero.

From Table 1 we notice that when $T = 500$ the ARBIC model has order 2 and that the ARMABIC model has order (2,0), but the curves in Figure 1 do not coincide. This may seem contradictory, since an ARMA$(p, 0)$ model is an AR(p) model. However, when fitting an AR model we use the Yule-Walker estimates, which are not identical to the maximum likelihood estimates that we employ when fitting an ARMA model.

Figure 1 and Table 1 about here.

For Figure 2 we generated sequences of length 100 and 500 from the mixed process $X_t = Y_t + Z_t$, where Y_t is the same data that was used for Figure 1 and $Z_t = 6\cos(t\pi/5 + \phi)$ with ϕ being uniformly distributed on $[-\pi, \pi]$. (The constant 6 was chosen so that $\text{var}(Y_t) \approx 2\,\text{var}(Z_t)$.) For both sample sizes LSPEC places an atom at $\pi/5$, the correct harmonic frequency. For the smaller sample size the mass of the atom was 8.05, and for the large sample size it was 8.43; the true mass at this frequency is $6^2/4 = 9$. As can be seen from Figure 2, the addition of an atom to the spectral distribution does not noticeably alter the

LSPEC estimate of the spectral density function. The ARMABIC estimate also changes little from the estimates shown in Figure 1, except for the noticeable smaller estimate for $t = 100$ for frequencies close to zero. However, the ARBIC estimate changes dramatically, not only for frequencies close to $\pi/5$, but also for frequencies close to zero. The ARAIC estimate changes slightly less when an atom is added; in particular, the peak near the location of the atom is narrower. However, the change is still quite large, and the ARAIC estimate even without the harmonic component is considerably more wiggly than the other estimates.

Figure 2 about here.

For Figure 3 we generated sequences of length 100 and 500 from the MA(4) process $Y_t = \epsilon_t - 0.3\epsilon_{t-1} - 0.6\epsilon_{t-2} - 0.3\epsilon_{t-3} + 0.6\epsilon_{t-4}$, where $\epsilon_t \sim_{\text{iid}} N(0,1)$. As can be seen, the LSPEC and ARMABIC estimates look more like the true spectral density function than the ARBIC estimate. For the ARAIC estimate, the difference is far more dramatic. In particular, from Table 1 below we note that the ARAIC models are of the order 5 and 14 for $T = 100$ and $T = 500$, respectively. This is typical of the erratic behavior of ARAIC estimates for MA models that cannot be well approximated by low order AR models.

Figure 3 about here.

Both our examples with spectral densities are ARMA processes. To investigate what would happen if the true underlying process is not an ARMA process we generated sequences of length 100 and 500 from the process $Y_t^{(4)} = 0.423Y_t^{(1)} + Y_t^{(3)}$, where $Y_t^{(1)}$ is the same data that was used for Figure 1 and $Y_t^{(3)}$ is the same data that was used for Figure 3. The constant 0.423 was chosen so that the contributions to the marginal variances of the two components were approximately equal. As can be seen from Figure 4, all estimates are quite good in this situation, except for the ARBIC estimate for the smaller sample size.

Figure 4 about here.

In the remainder of this section we focus on processes of the form

$$X_t = \sum_{j=1}^{p} R_j \cos(t\lambda_j + \phi_j) + \epsilon_t, \tag{5}$$

where R_j are positive constants, ϕ_j are independent and uniformly distributed on $[-\pi, \pi]$, and $\epsilon_t \sim_{\text{iid}} N(0,1)$.

First let $p = 0$. Then X_t is Gaussian white noise. In this case the LSPEC estimate should have one knot and no atoms, while the two AR estimates should have order zero and the two ARMA estimates should have order $(0,0)$. To find out how frequently this actually happens, we generated 100 samples of various sizes and applied the LSPEC, ARAIC and ARBIC

estimates to each sample. The results are summarized in Table 2. According to this table the ARBIC and ARMABIC estimates consistently pick the right model for large sample sizes, LSPEC picks the right model most of the time, while the ARAIC and ARMAAIC estimates are inconsistent.

Table 2 about here.

In Figure 5 we display LSPEC, ARBIC and ARMABIC estimates for several models generated from (5), all with $T = 500$. (The AIC estimates are extremely close to the BIC estimates.) Only the part of the spectrum near the true atoms is shown. In this figure we show either the estimated line spectrum or the estimated spectral density function. The actual values for p, R_p and λ_p are given in Table 3.

Table 3 and Figure 5 about here.

In Figure 5(a), the atom is not of the form $2\pi k/T$ with k an integer. Note that LSPEC handles this by estimating two atoms, one at $.304\pi$ and one at $.308\pi$, the two closest frequencies at which LSPEC may locate atoms. The true atoms for (b) and (c) of Figure 5 are of the form $2\pi k/T$ with k an integer. We note that LSPEC places both atoms in each of these two plots correctly. The ARBIC and ARMABIC estimates, on the other hand, 'lump' both atoms together in one peak in Figure 5(b), although the true atoms are separated by $3\times 2\pi/T$ in the corresponding model. Observe that ARBIC correctly fits twin peaks in Figure 5(c), in which the true atoms are separated by $6 \times 2\pi/T$, while ARMABIC still lumps both peaks together. If there are three consecutive atoms, as in Figure 5(d), LSPEC estimates the spectral distribution as having a density function. The two spectral density estimates are on the same scale in this figure. Here one could argue that ARBIC and ARMABIC are better than LSPEC, since the later underestimates the variance of the corresponding process.

To investigate how frequently the various estimates separate two atoms, we carried out the following simulation. We generated 100 series of length 500 from model (5) with $p = 2$, $R_1 = R_2 = R$, $\lambda_1 = .3\pi$ and $\lambda_2 = \lambda_1 + 2d\pi/500$ for various values of R and d. For LSPEC we counted how often it places atoms at λ_1 and λ_2. For ARAIC and ARBIC we counted how often $2\hat{f}((\lambda_1 + \lambda_2)/2) < \min(\hat{f}(\lambda_1), \hat{f}(\lambda_2))$. The results are summarized in Table 4 below.

Table 4 about here.

As can be seen from this table, LSPEC essentially always estimates two atoms when two atoms are present. For ARAIC to yield two peaks, the atoms must be at least about $6 \times 2\pi/T$ apart even when the signal to noise ratio is very large. For ARBIC the two atoms need to be even further apart. The ARMA procedures perform better; for them to yield two peaks, the atoms must be at least about $4 \times 2\pi/T$ apart. It should also be noted that

if ARBIC and ARAIC both estimate two atoms, then the two estimates are usually very similar, while the two ARMA estimates are virtually always identical in this set-up. Because of the convergence problems that we mentioned above for the ARMA procedures, when R was 500 a few times out of the 100 simulations all high order models produced errors, making it impossible for ARMAAIC and ARMABIC to score a "100" in these situations.

8. REAL EXAMPLES

Figure 6 shows the result of applying LSPEC, ARBIC and ARMABIC to the Canadian Lynx data, which has been studied extensively in the time series literature; see, for example, Campbell and Walker (1977), Tong (1977), Bhansali (1979) and Priestley (1981). (Actually, we applied the various estimates to the logarithm to the base ten of the data as has been done in previous analyses of this data.) An interesting issue involving this data is the existence of an atom at an angular frequency near $(24/114)\pi$, which corresponds to a period of 9.5 years. This cycle is usually explained as being due to a predator-prey relationship between the Canadian Lynx and the snowshoe hare, its most important source of food. When we applied LSPEC to this data we actually did obtain an atom at this frequency. The mass of this atom is about 0.089, and it explains 56% of the variability in the data. Both the ARMAAIC and the ARMABIC model are of order $(3, 3)$. As can be seen from Figure 6, these models agree with the LSPEC estimates with respect to the possible existence of an atom. The ARBIC model has order two, while the ARAIC model has order eleven.

Our second example involving real data is Wolf's sunspot data (Morris, 1977). The data consists of the annual average value of the daily index of the number of sunspots for the years 1755–1964. Again the discussion centers around the existence of an atom in the spectral distribution. Newton (1988) summarizes this as: "The basic property of these data is that there appear to be cyclic patterns but that these patterns are not perfectly cyclic." Here the LSPEC estimate, shown in Figure 7, has no atoms and five knots, thus suggesting that an absolutely continuous distribution fits the data. The ARAIC model has order eight, the ARBIC model has order two, the ARMAAIC model has order (8,1) and the ARMABIC model has order (3,4). Note that the ARBIC estimate is considerably smaller near zero than the other estimates, while the ARMABIC estimate has a considerably higher peak near the frequency $\pi/6$.

The final example involving real data is a series of length 600 of monthly water levels in Lake Erie from 1921 to 1970, which is one of the running examples in Newton (1988). The estimates for this example are shown in Figure 8. Here LSPEC assigns mass 0.4 to an atom at the angular frequency $\pi/6$, which corresponds to a period of one year. This atom explains about 22% of the variation in the series. The ARAIC and ARBIC estimates both have order 13 and the ARMAAIC and ARMABIC estimates both have order $(8, 5)$.

Figures 6–8 about here.

9. DISCUSSION

In light of the examples in Sections 7 and 8 and much additional experience with LSPEC, we are convinced that this methodology is of considerable practical value in data analysis. As can be seen from Figures 1–5, the LSPEC estimates of both the line spectrum and the spectral density function are fairly accurate, whether the true spectral distribution is discrete, continuous or mixed. In particular, when the true spectral distribution is mixed, the estimate of the spectral density function is not influenced by the discrete component of the spectral distribution (see Figure 2). Even if two atoms are close together, LSPEC estimates them as two separate atoms. If three or more atoms are close together (that is, at consecutive frequencies), however, then LSPEC typically assigns the corresponding mass to the spectral density function.

The performance of LSPEC when the true underlying spectrum is discrete is in sharp contrast to that of ARAIC and ARBIC. For the AR procedures to separate two atoms, they need to be at least about $6 \times 2\pi/T$ apart. Generally, in our experience, when the underlying spectral distribution is mixed, the AR estimates of the spectral density function near the atoms are rather poor and ARBIC performs worse than ARAIC. When one atom is present, ARMA estimators perform better than AR estimates, and they are comparable to LSPEC estimates. However when there is more than one atom, as is the case in Figure 5 (b)–(d) and Table 4, LSPEC performs much better than AR and ARMA procedures.

When the true spectral distribution is absolutely continuous, LSPEC is competitive with ARBIC and ARMABIC, while the ARAIC and ARMAAIC estimates are more wiggly. In particular, when the underlying process is an AR process, ARBIC, not unexpectedly, performs slightly better than LSPEC. However, if the underlying process cannot be well approximated by a low order AR process as in the case of Figure 3 and 4, then LSPEC and ARMABIC usually outperforms ARBIC.

A limitation of LSPEC is its ability to distinguish between very narrow peaks in the spectral density function, which might be only $2 \times 2\pi/T$ wide, and a few tightly clustered atoms. On the other hand, AR and ARMA estimates do not distinguish at all between peaks in the underlying spectral density function and atoms. Another current limitation of LSPEC is that the fitted atoms are restricted to the form $2\pi j/T$. If a true atom is not of this form, the LSPEC estimate typically has two atoms, one on either side of the true atom. Presumably, a refined version of LSPEC could be developed that would allow atoms at any frequency. In the mean time, however, we feel that the present LSPEC implementation is a useful addition to the toolbox for the analysis of time series.

Under suitable conditions, Kooperberg, Stone and Truong (1995) obtain the L_2 rate of convergence for a nonadaptive version of logspline spectral density estimation. This result lends theoretical support to LSPEC.

10. NUMERICAL IMPLEMENTATION

10.1. The Rao statistics

We elaborate here on the computation of the Rao statistic for the addition of a knot to a model. (For the addition of an atom all but one of the terms in equations comparable to (6)–(8) below are zero, so the computation of the corresponding Rao statistic is trivial.) Given a model with $K_c - 1$ knots $t_1, \ldots, t_{K_c-1}$, K_d atoms $a_1, \ldots, a_{K_d}$ and a potential knot at t_K^*, define a new basis function B_K^* by

$$B_K^*(\lambda) = \frac{3((\pi - t_1)^2 - (\pi - t_K^*)^2)}{2\pi}\lambda^2 + (\lambda - t_K^*)_+^3 - (\lambda - t_1)_+^3, \qquad 0 \le \lambda \le \pi;$$

here $x_+ = \max(0, x)$. Set

$$g_j = I_j \exp\left(-\sum_{k=1}^{K-1} \hat{\beta}_k B_k\left(\frac{2\pi j}{T}\right)\right), \qquad 1 \le j \le T/2,$$

where $\hat{\beta}_k$ are the maximum likelihood estimates of β_k for the model with $K-1$ parameters.

To compute the Rao statistic, the following new elements of the score function and Hessian need to be computed:

$$\mathbf{S}(\widehat{\boldsymbol{\beta}}_0)_K = \sum_{j=1}^{n}(g_j - 1)B_K^*\left(\frac{2\pi j}{T}\right), \tag{6}$$

$$\mathbf{H}(\widehat{\boldsymbol{\beta}}_0)_{KK} = \sum_{j=1}^{n} -g_j\left[B_K^*\left(\frac{2\pi j}{T}\right)\right]^2 \tag{7}$$

and

$$\mathbf{H}(\widehat{\boldsymbol{\beta}}_0)_{Kk} = \mathbf{H}(\widehat{\boldsymbol{\beta}}_0)_{kK} = \sum_{j=1}^{n} -g_j B_k\left(\frac{2\pi j}{T}\right) B_K^*\left(\frac{2\pi j}{T}\right), \qquad 1 \le k \le K_c - 1; \tag{8}$$

here $n = [T/2]$.

Before we compute the Rao statistic for any potential knot t_K^*, we first compute

$$a_{ml} = \sum_{j=1}^{m}\left(\frac{2\pi j}{T}\right)^l g_j, \qquad 1 \le m \le T/2 \text{ and } 0 \le l \le 6,$$

and

$$b_{mlk} = \sum_{j=1}^{m}\left(\frac{2\pi j}{T}\right)^l g_j B_k\left(\frac{2\pi j}{T}\right), \qquad 1 \le m \le n, \ 0 \le l \le 3 \text{ and } 1 \le k \le K_c - 1.$$

Since B_K^* is a cubic polynomial, on each of the three intervals $[0, \min(t_1, t_K^*)]$, $[\min(t_1, t_K^*), \max(t_1, t_K^*)]$, $[\max(t_1, t_K^*), \pi]$, we can compute all the new elements in $O(K)$ CPU-time using

the values of a_{ml} and b_{mlk}. (Note that $\mathbf{S}(\widehat{\boldsymbol{\beta}}_0)_k$ equals 0 unless $k = K$.) It follows by elementary matrix algebra (see Rao (1973, Problem 1.2.6)), that the Rao statistic equals

$$\frac{\left[\mathbf{S}(\widehat{\boldsymbol{\beta}}_0)_K\right]^2}{\mathbf{H}(\widehat{\boldsymbol{\beta}}_0)_{KK} - \mathbf{h}^t\mathbf{A}^{-1}\mathbf{h}},$$

where $\mathbf{h}$ is the column vector of length $K-1$ with the elements $\mathbf{H}(\widehat{\boldsymbol{\beta}}_0)_{Kk}$, $1 \le k \le K-1$, and $\mathbf{A}$ is the $(K-1)\times(K-1)$ matrix with the elements $\mathbf{H}(\widehat{\boldsymbol{\beta}}_0)_{jk}$, $1 \le j,k \le K-1$. Therefore, we can compute the Rao statistic for all frequencies $2\pi j/T$, $1 \le j \le T/2$, in $K^2T/4+O(KT)$ operations.

10.2. Quadratic approximations to the likelihood

Let Q be a quadratic polynomial on $\mathbb{R}^q$ having a negative definite Hessian matrix $\mathbf{H}$, and set $\mathbf{I} = -\mathbf{H}$. Also, let $\widehat{\boldsymbol{\beta}}$ maximize Q on $\mathbb{R}^q$ and let $\widehat{\boldsymbol{\beta}}_0 \in \mathbb{R}^q$. Then

$$0 = \nabla Q(\widehat{\boldsymbol{\beta}}) = \nabla Q(\widehat{\boldsymbol{\beta}}_0) + \mathbf{H}(\widehat{\boldsymbol{\beta}} - \widehat{\boldsymbol{\beta}}_0),$$

so $\widehat{\boldsymbol{\beta}} - \widehat{\boldsymbol{\beta}}_0 = \mathbf{I}^{-1}\nabla Q(\widehat{\boldsymbol{\beta}}_0)$, hence

$$\begin{aligned} Q(\widehat{\boldsymbol{\beta}}) &= Q(\widehat{\boldsymbol{\beta}}_0) + (\widehat{\boldsymbol{\beta}} - \widehat{\boldsymbol{\beta}}_0)^T\nabla Q(\widehat{\boldsymbol{\beta}}_0) + \tfrac{1}{2}(\widehat{\boldsymbol{\beta}} - \widehat{\boldsymbol{\beta}}_0)^T\mathbf{H}(\widehat{\boldsymbol{\beta}} - \widehat{\boldsymbol{\beta}}_0) \\ &= Q(\widehat{\boldsymbol{\beta}}_0) + \tfrac{1}{2}[\nabla Q(\widehat{\boldsymbol{\beta}}_0)]^T\mathbf{I}^{-1}\nabla Q(\widehat{\boldsymbol{\beta}}_0), \end{aligned}$$

and therefore

$$2[Q(\widehat{\boldsymbol{\beta}}) - Q(\widehat{\boldsymbol{\beta}}_0)] = [\nabla Q(\widehat{\boldsymbol{\beta}}_0)]^T\mathbf{I}^{-1}\nabla Q(\widehat{\boldsymbol{\beta}}_0). \tag{9}$$

Suppose now that $\widehat{\boldsymbol{\beta}}_0$ maximizes $Q(\boldsymbol{\beta})$ subject to the constraint that $\mathbf{A}\boldsymbol{\beta} = 0$, where $\mathbf{A}$ is a $p \times q$ matrix having rank p. Then $\mathbf{A}\widehat{\boldsymbol{\beta}}_0 = 0$. By the Lagrange multiplier theorem, there is a $\boldsymbol{\lambda} \in \mathbb{R}^p$ such that $\nabla Q(\widehat{\boldsymbol{\beta}}_0) = \mathbf{A}^T\boldsymbol{\lambda}$. It follows from (9) that

$$2[Q(\widehat{\boldsymbol{\beta}}) - Q(\widehat{\boldsymbol{\beta}}_0)] = \boldsymbol{\lambda}^T\mathbf{A}\mathbf{I}^{-1}\mathbf{A}^T\boldsymbol{\lambda}. \tag{10}$$

Moreover, $\widehat{\boldsymbol{\beta}} - \widehat{\boldsymbol{\beta}}_0 = \mathbf{I}^{-1}\mathbf{A}^T\boldsymbol{\lambda}$, so $\mathbf{A}\widehat{\boldsymbol{\beta}} = \mathbf{A}(\widehat{\boldsymbol{\beta}} - \widehat{\boldsymbol{\beta}}_0) = \mathbf{A}\mathbf{I}^{-1}\mathbf{A}^T\boldsymbol{\lambda}$. Thus, by(10),

$$2[Q(\widehat{\boldsymbol{\beta}}) - Q(\widehat{\boldsymbol{\beta}}_0)] = (\mathbf{A}\widehat{\boldsymbol{\beta}})^T(\mathbf{A}\mathbf{I}^{-1}\mathbf{A}^T)^{-1}(\mathbf{A}\widehat{\boldsymbol{\beta}}). \tag{11}$$

If Q is the quadratic approximation to the log-likelihood function at $\widehat{\boldsymbol{\beta}}_0$, then the right side of (9) is the Rao statistic. If Q is the quadratic approximation to the log-likelihood function at $\widehat{\boldsymbol{\beta}}$, then the right side of (11) is the square of the Wald statistic.

10.3. Number of basis functions

We stop the addition of basis functions when one of the following three conditions is satisfied:

- the number K of basis functions equals $K_{\max}$, where the default for $K_{\max}$ equals $\max(15, \min(4n^{1/5}, n/4, 30))$;
- $\hat{\ell}_K - \hat{\ell}_k < \frac{1}{2}(K-k) - 0.5$ for some k with $3 \leq k \leq K-3$, where $\hat{\ell}_k$ is the log-likelihood for the model with k basis functions (this suggests that adding more basis functions would waste CPU time since it is unlikely to yield an improved fit);
- the search algorithm, as described above, yields no possible position for a new basis function.

Note that the default value for $K_{\max}$ is somewhat arbitrary and mainly the result of experience. However, the power rate is somewhat motivated by the theoretical results in Kooperberg, Stone and Truong (1995), the $n/4$ bound prevents models for small data sets from being too large, while the constant upper bound prevents models for large data sets from being too large.

10.4. Numerical problems

Occasionally, there are numerical problems in carrying out the Newton-Raphson method when the LSPEC program is applied to a given time series. In particular, this happens when some $I_j/\bar{I}$ is very close to zero (of the order of magnitude of 10^{-30}), where $\bar{I}$ is the average value of I_j, $1 \leq j \leq n$. It can also happen if, for a fairly long sequence, the algorithm puts knots too close together. We overcome these numerical problems by the following fixes:

- the first time a numerical problem occurs, we restart the algorithm using I_j^* instead of I_j, where $I_j^* = I_j + 10^{-6}\bar{I}$;
- the kth time such a problem occurs, where $2 \leq k \leq 5$, we restart the algorithm but require that any knots that are added differ by at least $(k+1)2\pi/T$ from the knots already in the model;
- the sixth time the problem occurs, we terminate the program with an error message.

10.5. Autocovariances and random series

Let M be a large positive integer. We approximate the spectral distribution having spectral density function $\hat{f}_c$ by the discrete distribution having the line spectrum $\tilde{f}_c$, where

$$\tilde{f}_c(\lambda) = \begin{cases} \frac{\pi}{M}\hat{f}_c(\lambda) & \text{if } \lambda = j\pi/M \text{ with } |j| \le M-1, \\ \frac{\pi}{2M}\hat{f}_c(\lambda) & \text{if } \lambda = \pm\pi, \\ 0 & \text{otherwise.} \end{cases}$$

Note that (3) can be inverted to yield that $\gamma_c(u) = \int_{-\pi}^{\pi} \exp(iu\lambda) f_c(\lambda) d\lambda$. The fitted autocovariance function $\hat{\gamma}$ is defined as the autocovariance function having the line spectrum $\tilde{f} = \tilde{f}_c + \frac{T}{2\pi}\hat{f}_d$, which is given by

$$\hat{\gamma}(u) = \sum_{j=-M}^{M} \exp(iuj\pi/M)\tilde{f}_c(j\pi/M) + \frac{T}{\pi}\sum_{j=1}^{K_d} \exp(iua_j)\hat{f}_d(a_j);$$

see Isaacson and Keller (1966, Section 5.1). Using the fast Fourier transform, we can compute $\hat{\gamma}(u)$, $u = 0, \ldots, M$, in order $M \log M$ operations.

Let U_0 and U_j and V_j, $1 \le j \le M$, be independent normal random variables having mean zero and such that $\mathrm{var}(U_0) = \tilde{f}(0)$ and $\mathrm{var}(U_j) = \mathrm{var}(V_j) = \tilde{f}_c(j\pi/M)/2$, $1 \le j \le M$. Set $Z_0 = U_0$ and $Z_j = U_j + iV_j$ and $Z_{-j} = U_j - iV_j$, $1 \le j \le M$. Let R_j, $1 \le j \le K_d$, be independent non-negative random variables such that $2R_j^2/\hat{f}_d(a_j)$ has the χ^2 distribution with two degrees of freedom, and let S_j, $1 \le j \le K_d$, be independent random variables each having the uniform distribution on $[-\pi, \pi]$. Then

$$X_t^* = \sum_{j=1}^{K_d} R_j \cos(ta_j + S_j) + \sum_{j=-M}^{M} \exp(itj\pi/M) Z_j \tag{12}$$

is a real-valued Gaussian stationary time series with line spectrum $\tilde{f}$ and hence autocovariance function $\hat{\gamma}$ (see (1)). Using the fast Fourier transform, we can compute X_t^*, $t = 1 - M, \ldots, M$, in order $M \log M$ operations.

Alternatively, it is possible to set $R_j = 2\sqrt{\hat{f}_d(a_j)}$ in (1). In that case, X_t^* is a stationary time series that has the form of the mixed model studied in Mackisack and Poskitt (1990).

As the default in implementing the above formulas, we choose M to be the smallest power of 2 that is greater than or equal to the maximum of 1024 or W, where W is the maximum of the length of the original time series or the largest lag among the autocovariances or the length of the random time series requested.

10.6. Atoms with small mass

In order to prevent spurious atoms from being included in the estimated spectral distribution, LSPEC avoids entering an atom having mass less than $b_T\sigma^2$, where σ^2 is the variance of the

time series. In later stages, LSPEC will not add an atom at a frequency that was disallowed at an earlier stage because its mass was too small. Also, at the stepwise deletion stage, atoms having mass less than $b_T\sigma^2$ are automatically removed. The default value for b_T is motivated by the requirement that, when applied to Gaussian white noise, LSPEC should include an atom in the estimated spectral distribution at most five percent of the time.

Specifically, let X_i, $0 \le i \le T-1$, be independent random variables with variance σ^2. Then I_j, $1 \le j \le T/2$, are asymptotically independent exponential random variables with mean $\sigma^2/2\pi$. If LSPEC fits one atom, we expect that it is at the frequency corresponding to the largest value I^* of I_j. The estimate of f at this frequency would be I^* instead of $\sigma^2/2\pi$ and the estimate of f_d would be about $(2\pi I^* - \sigma^2)/T$. Thus b_T is determined by the requirement that

$$P\left(\frac{2\pi I^* - \sigma^2}{T} < b_T\sigma^2\right) = 0.95. \tag{13}$$

Here $2\pi I^*/(\sigma^2 T)$ has approximately the distribution of the maximum of n independent random variables each having the exponential distribution with mean one, where $n = [T/2]$ Thus (13) can be rewritten as $P(Y < Tb_T + 1) = 0.95^{1/n}$, where Y is a random variable having the exponential distribution with mean one. Therefore we get that

$$b_T = \frac{-\log(1 - 0.95^{1/n}) - 1}{T}.$$

In particular, $b_{100} \approx 0.059$ and $b_{500} \approx 0.015$.

ACKNOWLEDGMENTS

We wish to thank John Rice for some simulating conversations. Charles J. Stone was supported in part by National Science Foundation Grant DMS-920427. Young K. Truong was supported in part by a Research Council Grant from the University of North Carolina.

REFERENCES

Ansley, C. F. (1979) An algorithm for the exact likelihood of a mixed autoregressive-moving average process. *Biometrika* 66, 59–65.

Beamish, N. and Priestley, M. B. (1981) A study of autoregressive and window spectral estimation. *Appl. Statist.* 30, 41–58.

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language.* Pacific Grove, California: Wadsworth.

BELTRÃO, K. I. and BLOOMFIELD, P. (1987) Determining the bandwidth of a kernel spectrum estimate. *J. of Time Ser. Anal.* 8, 21–38.

BHANSALI, R. J. (1979) A mixed spectrum analysis of the Lynx data. *J. R. Statist. Soc., Ser. A* 142, 199–209.

DE BOOR, C. (1978) *A Practical Guide to Splines.* New York: Springer.

BRILLINGER, D. R. (1981) *Time Series, Data Analysis and Theory.* San Francisco: Holden-Day.

BROCKWELL, P. J. and DAVIS, R. A. (1991) *Time Series*: *Theory and Methods.* Second edition, New York: Springer.

CAMPBELL, M. J. and WALKER, A. M. (1977) A survey of statistical work on the MacKenzie River series of annual Canadian lynx trappings for the years 1821–1834, and a new analysis. *J. R. Statist. Soc., Ser. A* 140, 411–31.

CHAMBERS, J. M. and HASTIE, T. J. (1992) *Statistical Models in S.* Pacific Grove, California: Wadsworth.

FRANKE, J. and HÄRDLE, W. (1992) On bootstrapping kernel spectral estimates. *Ann. Statist.* 20, 121–45.

HURVICH, C. M. and BELTRÃO, K. I. (1990) Cross-validatory choice of a spectrum estimate and its connections with AIC. *J. Time Ser. Anal.* 11, 121–37.

ISAACSON, E. and KELLER, H. B. (1966) *Analysis of Numerical Methods.* New York: Wiley.

KOOPERBERG, C. and STONE, C. J. (1992) Logspline density estimation for censored data. *J. Comp. Graph. Statist.* 1, 301–28.

KOOPERBERG, C., STONE, C. J. and TRUONG, Y. K. (1995) Rate of convergence for logspline spectral density estimation. *J. Time Ser. Anal.* to appear.

MACKISACK, M. S. and POSKITT, D.S. (1990) Some properties of autoregressive estimates for processes with mixed spectra. *J. Time Ser. Anal.* 11, 325–37.

MORRIS, J. (1977) Forecasting the sunspot cycle. *J. Roy. Statist. Soc. Ser. A* 140, 437–47.

NEWTON, H. J. (1988) *TIMESLAB*: *A time series analysis laboratory.* Pacific Grove, California: Wadsworth.

PAWITAN, Y. and GANGOPADHYAY, A. K. (1991) Efficient bias corrected nonparametric spectral estimation. *Biometrika* 78, 825–32.

Pawitan, Y. and O'Sullivan F. (1994) Nonparametric spectral density estimation using Whittle likelihood. *JASA*, 89, 600–610.

Politis, D. N. and Romano, J. P. (1992) A general resampling scheme for triangular arrays of α-mixing random variables with application to the problem of spectral density estimation. *Ann. Statist.* 20, 1985–2007.

Priestley, M. B. (1962a) The analysis of stationary processes with mixed spectra–I. *J. R. Statist. Soc., Ser. B* 24, 214–33.

——— (1962b) The analysis of stationary processes with mixed spectra–II. *J. R. Statist. Soc., Ser. B* 24, 511–29.

——— (1981) *Spectral Analysis and Time Series.* London: Academic Press.

Rao, C. R. (1973) *Linear Statistical Inference and Its Applications* (2nd edn). New York: Wiley.

Stoica, P., Moses, R. L., Söderström, T. and Li, J. (1991) Optimal high-order Yule-Walker estimation of sinusoidal frequencies. *IEEE Trans. on Acoustics* 39, 1360–68.

Swanepoel, J. W. and van Wyk, J. W. J. (1986) The bootstrap applied to spectral density function estimation. *Biometrika*, 73, 135–42.

Tong, H. (1977) Some comments on the Canadian Lynx data. *J. R. Statist. Soc., Ser. A* 140, 432–36.

Wahba, G. (1980) Automatic smoothing of the log periodogram. *J. Amer. Statist. Assoc.* 75, 122–32.

Whittle, P. (1961) Gaussian estimation in stationary time series. *Bull. Int. Statist. Inst.* 39, 105–30.

Department of Statistics
University of Washington
Seattle, Washington 98195

Department of Statistics
University of California
Berkeley, California 94720

School of Public Health
Department of Biostatistics
University of North Carolina
Chapel Hill, North Carolina 27599-7400

APPENDIX: DOCUMENTATION OF S FUNCTIONS

lspec.plot Logspline Spectral Estimation **lspec.plot**

DESCRIPTION

lspec.plot: plots a spectral function fitted with lspec.fit.

USAGE

```
lspec.plot(fit, what = "b", n, add = F, ...)
```

ARGUMENTS

`fit:`	a list like the output from lspec.fit.
`what:`	what should be plotted: b (spectral density and line spectrum superimposed), d (spectral density function), l (line spectrum) or p (spectral distribution function).
`n:`	the number of equally spaced points at which to plot the fit. Default is max(100, fit$sample).
`add:`	indicate that the plot should be added to an existing plot.
`...:`	all regular plotting options as desired.

VALUE

This function produces a plot of an lspec fit at n equally spaced points from 0 to π (from $-\pi$ to π if what = "p"). Use xlim=c(from,to) to change the range of these points. If add = T, xlim is taken as the limits of the plot to which the present curve is added. If what = "p", the plotting range cannot extend beyond the interval $[-\pi, \pi]$.

lspec.summary Logspline Spectral Estimation **lspec.summary**

DESCRIPTION

lspec.summary: summarizes an lspec fit.

USAGE

```
lspec.summary(fit)
```

ARGUMENTS

`fit:`	a list like the output from lspec.fit.

VALUE

This function produces a printed summary of an lspec.fit.

LSPEC Logspline Spectral Estimation **LSPEC**

DESCRIPTION

Autocorrelations, autocovariances, spectral distributions and random time series from a model fitted with lspec.fit.

USAGE

```
clspec(lag, fit, cov = T, mm = <<see below>>)
plspec(freq, fit, mm = <<see below >>)
rlspec(n, fit, mean = 0, cosfixed = F, mm = <<see below >>)
```

ARGUMENTS

`lag:`	vector of integer-valued lags for which the autocorrelations or autocorrelations are to be computed.
`cov:`	compute autocovariances (T) or autocorrelations (F).
`freq:`	vector of frequencies. Frequencies should be between $-\pi$ and π.
`n:`	length of the random time series to be generated.
`mean:`	mean level of the time series to be generated.
`fit:`	a list like the output from lspec.fit.
`cosfixed:`	indicates that the data should be generated from a model with constant harmonic components rather than a true Gaussian time series.
`mm:`	number of points used in integration and the fft. Default is the smallest power of two larger than max(fit$sample, n, max(lag), 1024).

VALUE

autocovariances or autocorrelations (clspec), spectral distribution (plspec) or a random time series (rlspec).

dlspec Logspline Spectral Estimation **dlspec**

DESCRIPTION

spectral density function and line spectrum from a model fitted with lspec.fit.

USAGE

```
dlspec(freq, fit)
```

ARGUMENTS

`freq:`	vector of frequencies.
`fit:`	a list like the output from lspec.fit.

VALUE

`d:`	the spectral density function evaluated at the vector of frequencies.
`modfreq:`	modified frequencies of the form $2\pi j/T$ that are close to the frequencies that were requested.
`m:`	mass of the line spectrum at the modified frequencies.

DESCRIPTION

lspec.fit: fit a logspline spectral model.

USAGE

```
lspec.fit(data, period, penalty = <<see below>>, minmass = <<
see below>>, knots, maxknots, atoms, maxatoms, maxdim = <<see
below>>, odd = F, updown = 3)
```

ARGUMENTS

`data:` time series (exactly one of data and period should be specified). If data is specified, lspec.fit first computes the modulus of the fast Fourier transform of the series, resulting in a raw periodogram of length floor(length(data)/2).

`period:` value of the periodogram for a time series at frequencies $2\pi j/T$, for $1 \leq j \leq T/2$. If period is specified, odd should indicate whether T is odd (odd = T) or even (odd = F). Exactly one of data and period should be specified.

`penalty:` the parameter to be used in the AIC criterion. The method chooses the number of knots that minimizes $-2\times$loglikelihood+penalty$\times$(number of knots). The default is to use penalty=log(period) as in BIC.

`minmass:` threshold value for atoms. No atoms having smaller mass than minmass are included in the model. If minmass takes its default value, in 95% of the samples, when data is Gaussian white noise, the model will not contain an atom.

`knots:` ordered vector of values, which forces the method to start with these knots. If knots is not specified, the program starts with one knot at zero and then employs stepwise addition of knots and atoms.

`maxknots:` maximum number of knots allowed in the model. Does not need to be specified, since the program has a default for maxdim and the number of dimensions equals the number of knots plus the number of atoms. If maxknots = 1 the fitted spectral density function is constant.

`atoms:` ordered vector of values, which forces the method to start with discrete components at these frequencies. The values of atoms are rounded to the nearest multiple of $2\pi/$(length(data)). If atoms is not specified, the program starts with no atoms and then performs stepwise addition of knots and atoms.

`maxatoms:` maximum number of discrete components allowed in the model. Does not need to be specified, since the program has a default for maxdim and the number of dimensions equals the number of knots plus the number of atoms. If maxatoms = 0, a continuous spectral distribution is fit.

`maxdim:` maximum number of basis functions allowed in the model (default is $\max(15, 4 \times \text{length(period)}^{0.2})$).

`odd:` see period. If period is not specified, odd is not relevant.

`updown:` the maximal number of times that lspec.fit should go through a cycle of stepwise addition and stepwise deletion until a stable solution is reached.

VALUE

The output is organized to serve as input for lspec.summary, clspec, dlspec, plspec, rlspec and lspec.plot.

The function returns a list with the following members:

`call:` the command that was executed.

`thetap:` coefficients of the polynomial part of the spline.

`nknots:` the number of knots that were retained.

`knots:` vector of the locations of the knots in the model. Only the knots that were retained are in this vector.

`thetak:` coefficients of the knot part of the spline. The k-th coefficient is the coefficient of $(\text{x-t(k)})_+^3$.

`natoms:` the number of atoms that were retained.

`atoms:` vector of the locations of the atoms in the logspline model. Only the lines that were retained are in this vector.

`mass:` The k-th coefficient is the mass at atom[k].

`logl:` the log-likelihood of the model.

`penalty:` the penalty that was used.

`minmass:` the minimum mass for an atom that was allowed.

`sample:` the sample size that was used, either computed as length(data) or as $2\times$length(period) (odd=F) or $2\times$length(period)+1 (odd=T).

`updown:` the actual number of times that lspec.fit went through a cycle of stepwise addition and stepwise deletion until a stable solution was reached or minus the number of times that lspec.fit went through a cycle of stepwise addition and stepwise deletion until it decided to quit.

TABLE 1

Orders for the AR and ARMA estimates and number of atoms and knots for the LSPEC estimates for the examples in Figures 1-4.

	LSPEC		ARAIC (not shown)	ARBIC	ARMAAIC (not shown)		ARMABIC	
	atoms	knots	p	p	p	q	p	q
Figure 1, left side	0	2	6	2	6	0	1	1
Figure 1, right side	0	3	3	2	2	1	2	0
Figure 2, left side	1	2	5	2	5	2	5	2
Figure 2, right side	1	3	20	6	4	3	4	3
Figure 3, left side	0	4	5	3	3	4	0	4
Figure 3, right side	0	7	14	8	1	4	1	4
Figure 4, left side	0	5	8	1	1	4	1	4
Figure 4, right side	0	6	15	5	8	3	2	4

TABLE 2

Number of times that the correct model is estimated when the true process is Gaussian white noise.

Sample size	LSPEC	ARAIC	ARBIC	ARMAAIC	ARMABIC
50	76	71	93	54	84
100	90	77	99	48	97
500	90	74	100	42	98
1000	92	70	100	48	100

TABLE 3

Actual values for p, R_p and λ_p in Figure 5.

Figure	p	R_1	λ_1	R_2	λ_2	R_3	λ_3
(a)	1	12	$.3056\pi$				
(b)	2	6	$.300\pi$	6	$.312\pi$		
(c)	2	6	$.292\pi$	6	$.316\pi$		
(d)	3	4	$.300\pi$	4	$.304\pi$	4	$.308\pi$

TABLE 4
How often the various estimates separate two nearby atoms.

	LSPEC			ARAIC			ARBIC			ARMAAIC			ARMABIC		
d	$R=2$	12	500	2	12	500	2	12	500	2	12	500	2	12	500
1	88	99	100	0	0	3	0	0	2	0	0	7	0	0	7
2	100	100	100	0	0	10	0	0	6	0	0	36	0	0	36
3	100	100	100	0	0	20	0	0	12	0	18	68	0	17	68
4	100	100	100	0	21	31	0	13	18	0	65	79	0	65	79
5	100	100	100	0	44	50	0	20	27	0	81	88	0	81	88
6	100	100	100	0	86	95	0	29	30	1	97	95	1	98	95
7	100	100	100	0	99	100	0	51	45	3	98	93	1	99	93
8	100	100	100	0	100	100	0	72	61	7	95	97	6	99	97
9	100	100	100	4	100	100	0	95	93	18	100	97	15	100	97
10	100	100	100	46	100	100	0	99	97	38	100	96	33	100	96

FIGURE CAPTIONS

Figure 1. Estimated spectral density functions for samples from an AR(3) process; left T = 100, right T = 500; solid = LSPEC, short dashed = ARBIC, long dashed = ARMABIC, dotted = True.

Figure 2. Estimated spectral density functions for samples from a mixed process; left T = 100, right T = 500; solid = LSPEC, short dashed = ARBIC, long dashed = ARMABIC, dotted = True. Atoms are indicated by spikes starting at the height of the spectral density function.

Figure 3. Estimated spectral density functions for samples from an MA(4) process; left T = 100, right T = 500; solid = LSPEC, short dashed = ARBIC, long dashed = ARMABIC, dotted = True.

Figure 4. Estimated spectral density functions for samples from an AR(3) process mixed with an MA(4) process; left T = 100, right T = 500; solid = LSPEC, short dashed = ARBIC, long dashed = ARMABIC, dotted = True.

Figure 5. Estimated line spectrum (LSPEC, solid) and spectral density functions (ARBIC, short dashed and ARMABIC, long dashed) for various samples of size 500 from harmonic processes with white Gaussian noise.

Figure 6. Estimated spectral density functions for the Canadian lynx data (T = 114); solid = LSPEC, short dashed = ARBIC, long dashed = ARMABIC.

Figure 7. Estimated spectral density functions for the sunspot data (T = 215); solid = LSPEC, short dashed = ARBIC, long dashed = ARMABIC.

Figure 8. Estimated spectral density functions for the Lake Erie data (T = 600); solid = LSPEC, short dashed = ARBIC, long dashed = ARMABIC.

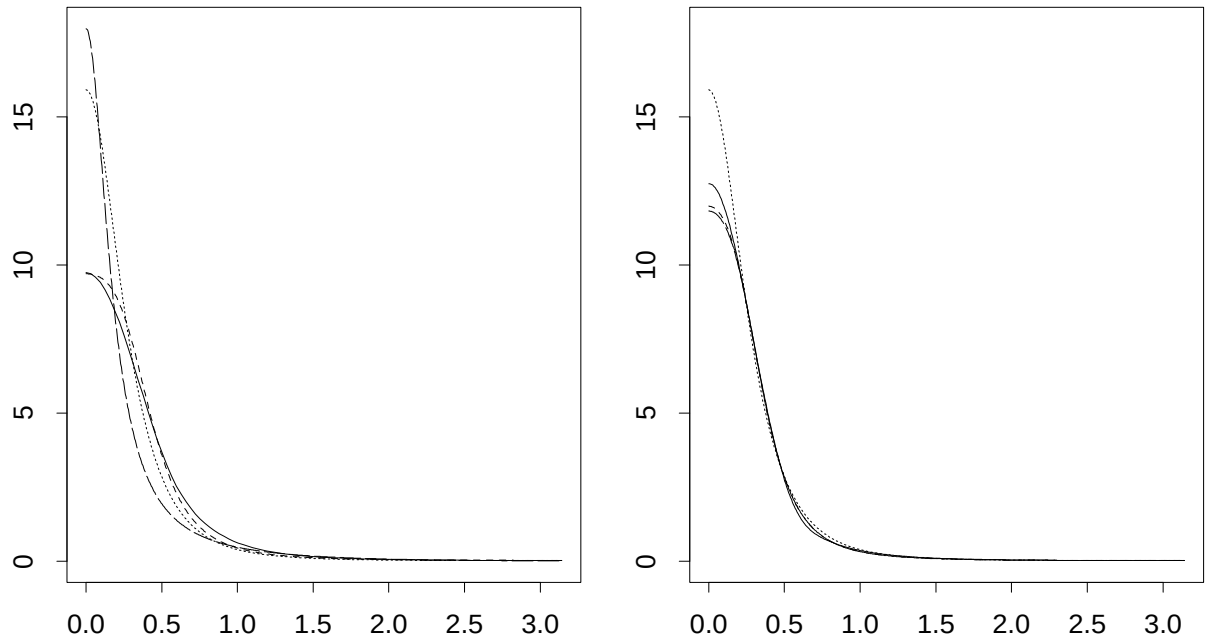

Figure 1. Estimated spectral density functions for samples from an AR(3) process; left T=100, right T=500; solid=LSPEC, short dashed=ARBIC, long dashed=ARMABIC, dotted=True.

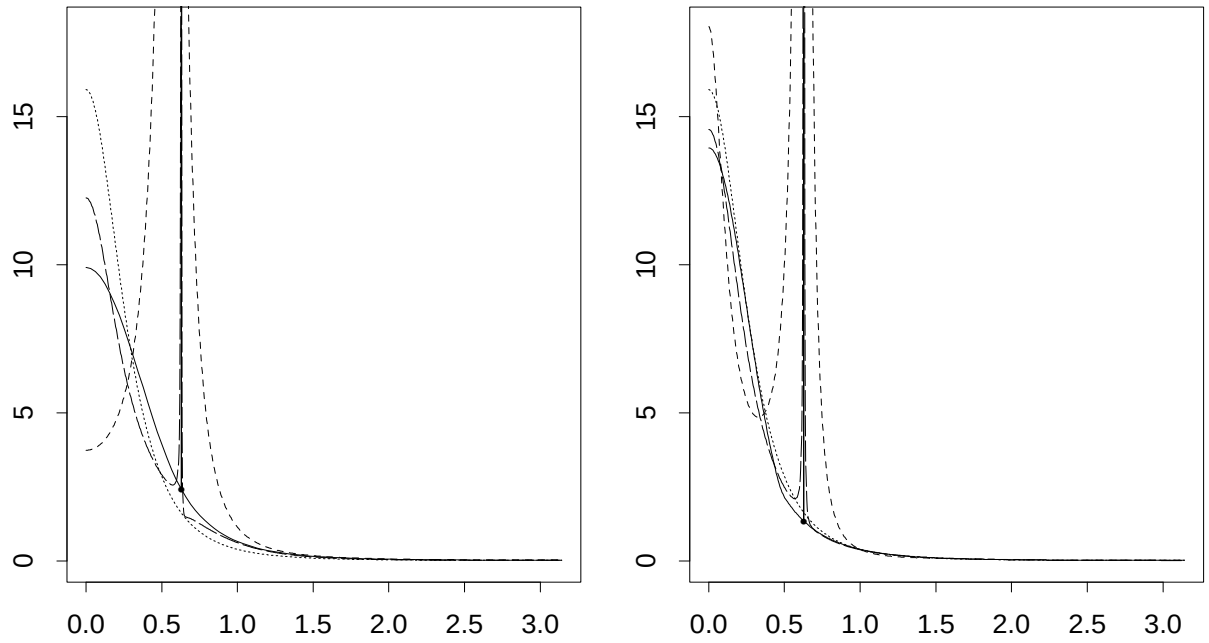

Figure 2. Estimated spectral density functions for samples from a mixed process; left T=100, right T=500; solid=LSPEC, short dashed=ARBIC, long dashed=ARMABIC, dotted=True. Atoms are indicated by spikes starting at the height of the spectral density function.

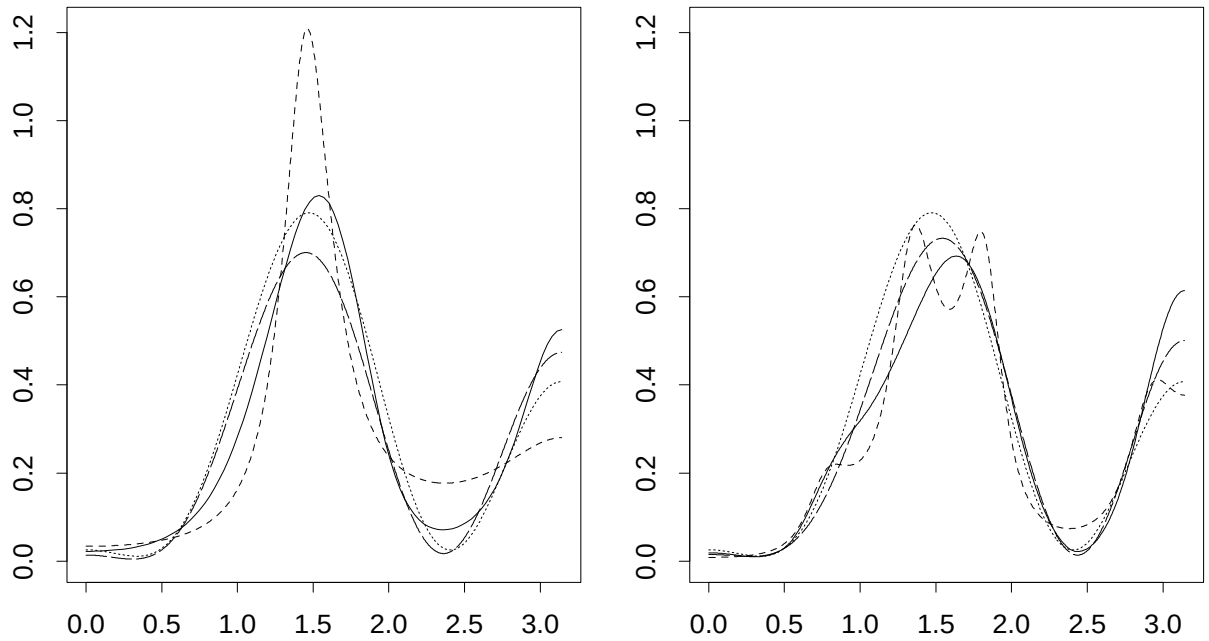

Figure 3. Estimated spectral density functions for samples from an MA(4) process; left T=100, right T=500; solid=LSPEC, short dashed=ARBIC, long dashed=ARMABIC, dotted=True.

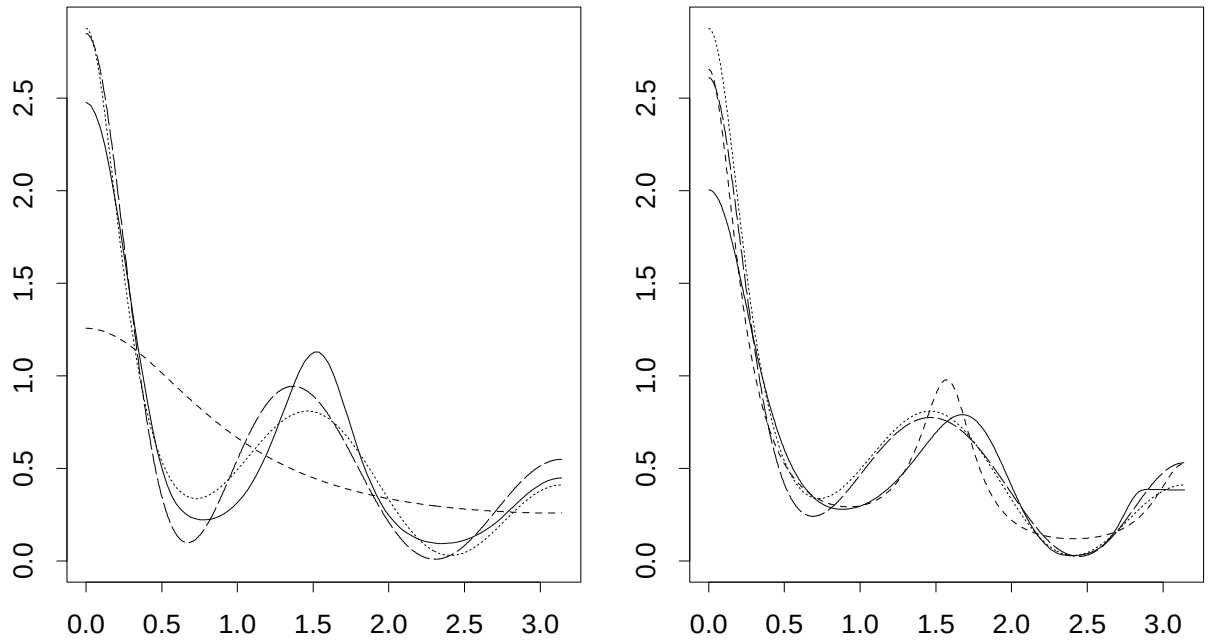

Figure 4. Estimated spectral density functions for samples from an AR(3) process mixed with an MA(4) process; left T=100, right T=500; solid=LSPEC, short dashed=ARBIC, long dashed=ARMABIC, dotted=True.

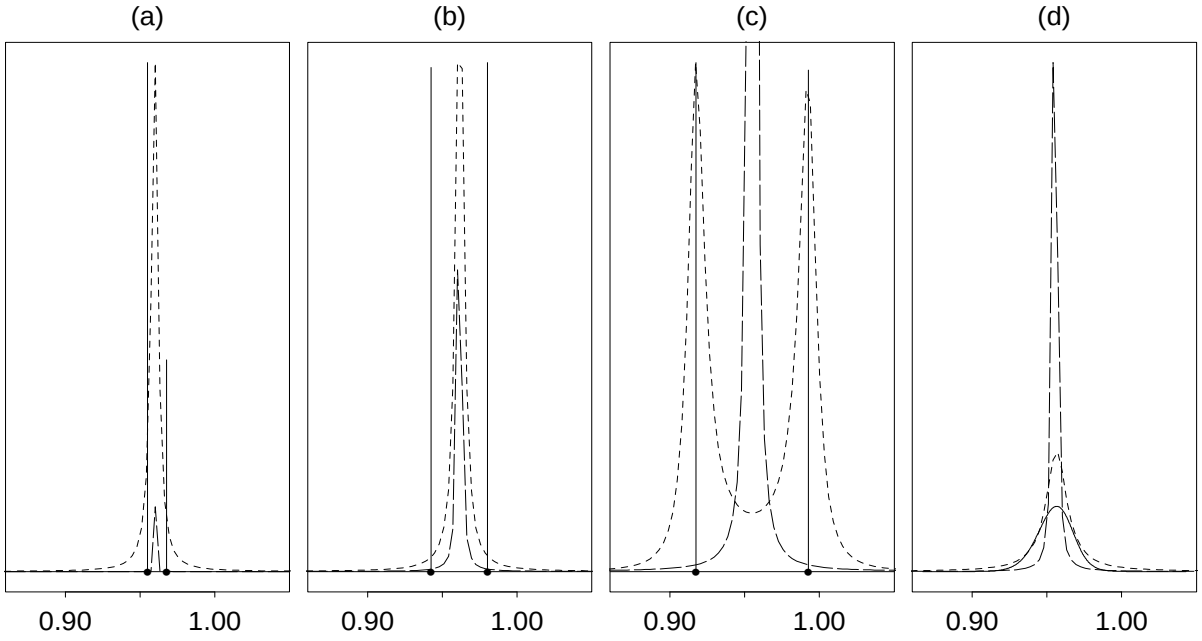

Figure 5. Estimated line spectrum (LSPEC, solid) and spectral density functions (ARBIC, short dashed and ARMABIC long dashed) for various samples of size 500 from harmonic processes with white Gaussian noise.

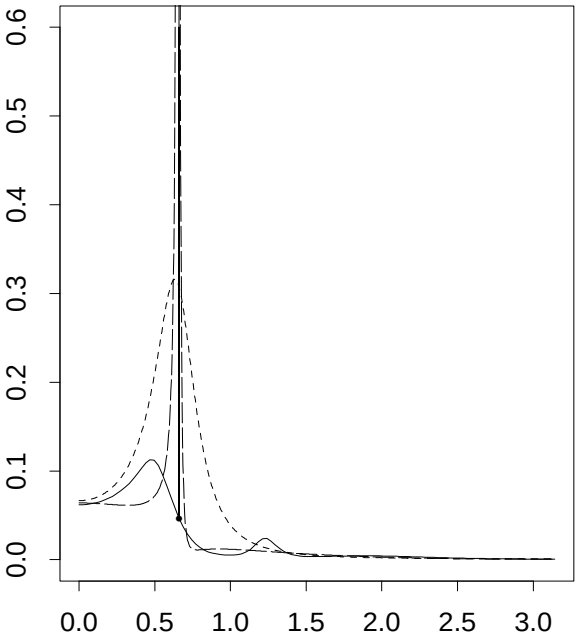

Figure 6. Estimated spectral density functions for the Canadian lynx data (T=114); solid=LSPEC, short dashed=ARBIC, long dashed=ARMABIC.

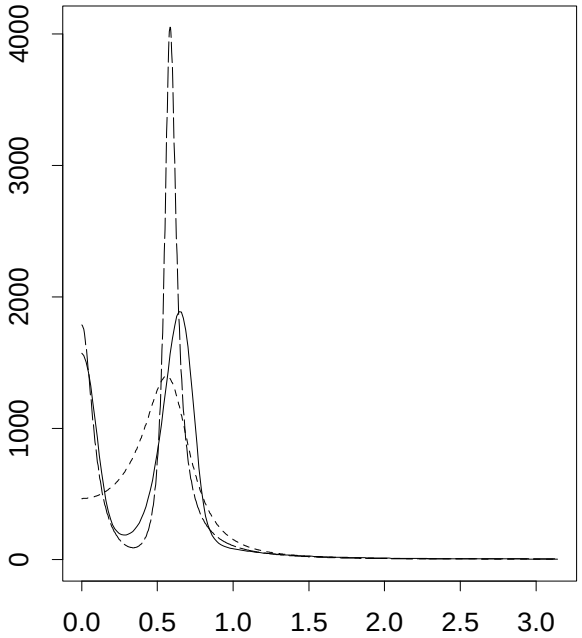

Figure 7. Estimated spectral density functions for the sunspot data (T=215); solid=LSPEC, short dashed=ARBIC, long dashed=ARMABIC.

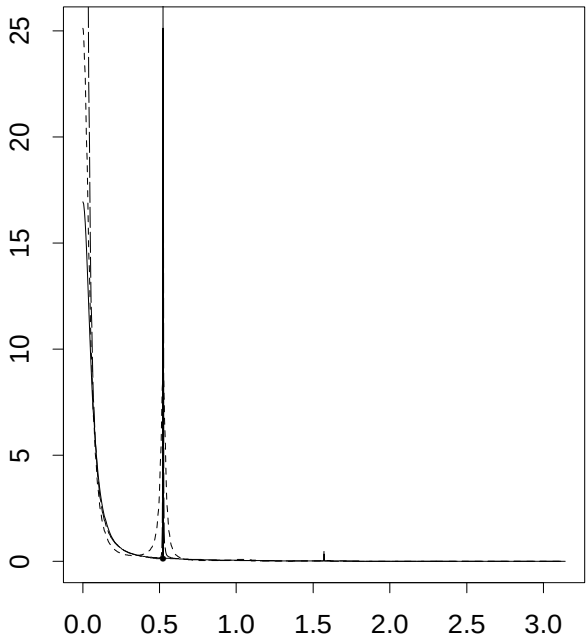

Figure 8. Estimated spectral density functions for the Lake Erie data (T=600); solid=LSPEC, short dashed=ARBIC, long dashed=ARMABIC.