Periodic Levinson-Durbin algorithm for entropy maximization

Georgi N. Boshnakov[a], Sophie Lambert-Lacroix[b]

[a] *School of Mathematics*
The University of Manchester
Oxford Road
Manchester M13 9PL, UK
[b] *UJF-Grenoble 1 / CNRS / UPMF / TIMC-IMAG*
UMR 5525, Grenoble, F-38041, France

Abstract

One gives a recursive algorithm for the computation of the first and second order derivatives of the entropy of a periodic autoregressive process with respect to the autocovariances. It is an extension of the periodic Levinson-Durbin algorithm. The algorithm has been developed for use at one of the steps of an entropy maximization method developed by the authors. Numerical examples of entropy maximization by that method are given. An implementation of the algorithm is available as an R package.

Keywords: Maximum entropy method, partial autocorrelation, periodically correlated processes, periodic Levinson-Durbin algorithm.

1. Introduction

The class of periodically correlated processes (pc-processes) introduced by Gladishev [8] is useful in many applications, see Hurd and Miamee [11] for a thorough exposition of the theory, Franses and Paap [7] for economic applications, Serpedin et al [17] for a comprehensive bibliography, and Hindrayanto et al [9] for state space modelling.

The maximum entropy principle provides an appealing framework for the specification of complete models from partial information. This method was introduced to stationary time series by Burg in the influential works [4, 5]. Given a contiguous set of autocovariances for lags $0, \ldots, p$, the maximum

entropy solution is an autoregressive process of order p with those autocovariances. In this case the problem is linear and the solution can be obtained by solving the Yule-Walker equations with the Levinson-Durbin algorithm. This result holds in the multivariate case as well. Lambert-Lacroix [13] generalised this result to pc-processes. Deep results on this and related problems have been obtained by Alpay et al [1] and Castro and Girardin [6].

When the lags are not contiguous the problem is, in general, non-linear but the solution is still an autoregression of order equal to the maximum specified lag. For univariate stationary processes, this particular case was studied by Politis [14] and the case of general gaps by Rozario and Papoulis [15]. A method for the solution of the maximum entropy problem for pc-processes in the case of general gap patterns has been developed in [3].

The entropy rate is a very complicated function of the autocovariances. It is hardly possible to write down useful expressions for it and its derivatives with respect to the non-specified autocovariances for general gap patterns. The periodic Levinson-Durbin algorithm (see [16] or [13]) can be used to calculate the entropy rate. For gradient and Newton-type maximization methods derivatives are also needed. In this paper we develop recursions for the first and second order derivatives of the entropy rate. We give also numerical examples that illustrate the behavior of our method. The R programs implementing the algorithm presented here and the maximum entropy method of [3] are available as the R package `pcme` [2].

The paper is organized as follows. Section 2 presents some basic results about maximum entropy for periodically correlated processes. Section 3 gives the algorithm for the calculation of the gradient and Hessian of the entropy. Numerical results illustrating the maximization of the entropy are presented in Section 4. Positive semidefinite (p.s.d.) solutions are discussed in Section 5.

2. The maximum entropy for periodically correlated processes

Let $\mathbb{N}$ be the set of the non-negative integers. A zero-mean process $\{X_t, t \in \mathbb{N} \setminus \{0\}\}$ is periodically correlated of period T if its autocovariance function $R(u,v) = E\left\{X_u \overline{X_v}\right\}$ is T-periodic, i.e.

$$R(u+T, v+T) = R(u,v), \qquad \text{for all } (u,v) \in \mathbb{N}^2, \tag{1}$$

(see [8], [10],[11]). It is convenient to think about the autocovariances in terms of the *seasons* $t = 1, \ldots, T$ and the *lags* $k \in \mathbb{N}$. Each pair $(u,v) \in \mathbb{N}^2$

may be represented as $(u, v) = (mT + t, mT + t - k)$ for some $t \in \{1, \ldots, T\}$, $m \in \mathbb{N}$, and integer k. From Equation (1) it follows that $R(mT+t, mT+t-k)$ does not depend on m. So, we may introduce the notation

$$R_t(k) = R(mT + t, mT + t - k), \qquad t \in \{1, \ldots, T\},\ m \in \mathbb{N},\ k\text{—integer}.$$

Moreover, it is sufficient to consider $R_t(k)$ for $k \geq 0$. Indeed, if $u - v = k < 0$, i.e. $v > u$, then $(v, u) = (m_1 T + s, m_1 T + s - |k|)$ for some $s \in \{1, \ldots, T\}$ and the value of $R(u, v)$ can be obtained from the identity $R(u, v) = \overline{R(v, u)} = \overline{R_s(|k|)}$. Similar notation is used by other authors, see [10].

To illustrate this notation, consider a monthly pc-process started in January 2000. Let $u = 13$ (January 2001) and $v = 11$ (November 2000). We have $T = 12$, $(u, v) = (1 * 12 + 1, 1 * 12 + 1 - 2)$ and $(v, u) = (0 * 12 + 11, 0 * 12 + 11 - (-2))$. So, $R(u, v) = R_1(2)$ and $R(v, u) = R_{11}(-2)$. On the other hand, $R_{11}(-2) = R(v, u) = \overline{R(u, v)} = \overline{R_1(2)}$.

If t is one of the seasons, $1, \ldots, T$, and k is a non-negative integer lag, then (t, k) will be called a *season-lag pair*. The T functions $R_1(\cdot)$, $\ldots$, $R_T(\cdot)$, considered as functions on $\mathbb{N}$, completely parameterise the second order structure of the pc-process in the sense that for each (u, v) there is exactly one season-lag pair (t, k) such that $R(u, v) = R_t(k)$ (if $u \geq v$) or $R(u, v) = \overline{R_t(k)}$ (if $u < v$). In other words, the doubly indexed sequence $\{R_t(k)\}$, $t \in \{1, \ldots, T\}$, $k \in \mathbb{N}$, enumerates the autocovariances in a non-redundant way. An equivalent parameterisation is given by the partial autocorrelations (pacf) $\{\beta_t(k)\}$, $t = 1, \ldots, T$, $k \in \mathbb{N}$ (see [13] for details).

Let I be a set of season-lag pairs and $K = \{R_t(k)\}_{(t,k)\in I}$ be a sequence defined on I. Let Γ be the set of all periodic autocovariance sequences whose values coincide with $R_t(k)$ for $(t, k) \in I$ (Γ may be empty). Each element of Γ is a completion (or extension) of K. The maximum entropy extension is the one whose entropy rate is maximal in Γ. The maximum entropy problem can be defined as follows.

ME(K, I) **problem** Given a set I of season-lag pairs and a sequence K defined on I, find the completion of K whose entropy rate is maximal or show that such a completion does not exist. In [3], we develop a method for the solution of the maximum entropy problem for arbitrary patterns of the set I on which the autocovariances are given. The method involves maximization of the entropy on season-lags sets of the form

$$E_c(I) = \{(t, k) | t = 1, \ldots, T,\ k = 0, \ldots, p_t\},$$

where $(p_1, \ldots, p_T)$ are the smallest non-negative integers satisfying the constraints $p_1 \leq p_T+1$ and $p_t \leq p_{t-1}+1$ for $t = 2, \ldots, T$, and such that $E_c(I) \supseteq I$. We refer the elements of $E_c(I) \backslash I$ as *gaps* since these are season-lag pairs with non-specified values in K.

To give the entropy rate definition we need additional notation. Let $\{X_t\}$ be a pc-process and let $v_t(k)$ be the variance of the prediction error of X_t in terms of the k previous values $X_{t-1}, \ldots, X_{t-k}$. Then for any given $t \in \{1, \ldots, T\}$ the sequence $\{v_{mT+t}(mT+t-1)\}_{m=1}^{\infty}$ is convergent as $m \to \infty$ since it is monotonically decreasing and bounded from below by 0. Let

$$\sigma_t^2 = \lim_{m \to \infty} v_{mT+t}(mT + t - 1), \qquad t = 1, \ldots, T. \tag{2}$$

An expression for σ_t^2 in terms of the partial autocorrelations is (see [13])

$$\sigma_t^2 = R_t(0) \prod_{n=1}^{\infty} (1 - \|\beta_t(n)\|^2), \qquad t = 1, \ldots, T.$$

It can be shown ([12]: p. 119) that for a Gaussian not locally deterministic pc-process X the entropy rate is equal to

$$h(X) = \frac{1}{2} \log(2\pi e) + \frac{1}{2T} \sum_{t=1}^{T} \log \sigma_t^2,$$

where $\sigma_t^2 > 0$ for $t = 1, \ldots, T$. Since we are considering only second order properties and the first term is a constant, we can define the entropy rate of a pc-process with autocovariance sequence R by

$$h(R) = \frac{1}{T} \sum_{t=1}^{T} \log \sigma_t^2. \tag{3}$$

If σ_t^2 is equal to 0 for some t (i.e. R is p.s.d.), then the entropy is defined to be $-\infty$. This convention means that if the sequence K has only positive semi-definite (p.s.d.) completions, then any one of them can be taken as the solution of the ME(K, I) problem. In such situations our algorithm picks up a p.s.d. completion with certain extremal properties, see Section 5 for a discussion and example.

3. PLD algorithm

Here we give formulae for the calculation of the elements of the gradient and the Hessian matrix of the entropy rate which are needed for the Newton-Raphson's algorithm. We denote by $\sigma_k^{f2}(n)$ and $\sigma_k^{b2}(n)$ the variances of the nth-order forward and backward partial innovations $\varepsilon_k^f(n)$ and $\varepsilon_k^b(n)$, defined by

$$\varepsilon_k^f(n) = \sum_{j=0}^{n} a_k^f(n,j) X_{k-j}, \quad a_k^f(n,0) = 1,$$
$$\varepsilon_k^b(n) = \sum_{j=0}^{n} a_k^b(n,j) X_{k-n+j}, \quad a_k^b(n,0) = 1.$$

where the filters $\{a_k^f(n,\cdot)\}$ and $\{a_k^b(n,\cdot)\}$ can be determined recursively in n, see below. So, $\sigma_k^{f2}(n)$ replaces the notation $v_k(n)$ for the variance of the prediction error introduced in Section 2. With this notation the expression for the entropy rate is

$$h(R) = \frac{1}{T} \sum_{k=1}^{T} \log \sigma_k^{f2}(p_k).$$

Its first and second derivatives are

$$\begin{aligned} \frac{\partial h(R)}{\partial R_t(l)} &= \frac{1}{T} \sum_{k=1}^{T} \frac{\frac{\partial \sigma_k^{f2}(p_k)}{\partial R_t(l)}}{\sigma_k^{f2}(p_k)}, \\ \frac{\partial^2 h(R)}{\partial R_t(l) \partial R_s(m)} &= \frac{1}{T} \sum_{k=1}^{T} \left[\frac{\frac{\partial^2 \sigma_k^{f2}(p_k)}{\partial R_t(l) \partial R_s(m)}}{\sigma_k^{f2}(p_k)} - \frac{\frac{\partial \sigma_k^{f2}(p_k)}{\partial R_t(l)} \frac{\partial \sigma_k^{f2}(p_k)}{\partial R_s(m)}}{(\sigma_k^{f2}(p_k))^2} \right], \end{aligned}$$

for gaps (t,l) and (s,m), i.e. $(t,l) \in E_c(I) \setminus I$ and $(s,m) \in E_c(I) \setminus I$.

So, we need to compute the first and second derivatives of the variances of the innovations with respect to autocovariances corresponding to gaps. Analytic closed form expressions for them in terms of the autocovariances are not available but they can be calculated recursively using the periodic Levinson-Durbin (PLD) algorithm (see [13]). The PLD algorithm computes, besides the variances, the filters $\{a_k^f(n,\cdot)\}$ and $\{a_k^b(n,\cdot)\}$ which are also needed for the calculation of the derivatives.

We differentiate the PLD equations to develop PLD-type recursions for the first and second order derivatives. In the formulae below we use the convention $\sum_{j=1}^{0} \ldots = 0$. Also, when the subscript $k-1$ is equal to 0, it is replaced by T. $\delta_{i,j}$ is the Kronecker symbol, that is $\delta_{i,j} = 1$ if $i = j$, $\delta_{i,j} = 0$ otherwise. We denote by $t[T]$ the integer r_t in $[1, \ldots, T]$ such that $t = q_t T + r_t$ and $q_t \in Z$.

Algorithm 1 summarizes the calculations of the derivatives. It does not include formulae since many of them are quite long. The formulae are given in the appendix and referred to in Algorithm 1.

ALGORITHM **1.** *Outline of the calculations of first and second order derivatives based on the PLD algorithm. An entry like Equations XX stands for the group of equations numbered (XXa), (XXb), ..., while Equation XX stands for a single equation.*

for $k = 1$ **to** T **do** `// Initialisation step`
  Equation (A.1)
  for $t = 1$ **to** T, *and* $l \in \tau_g$ **do**
    Equation (A.2)
    for $s = 1$ **to** T, *and* $m \in \tau_g$ **do**
      Equation (A.3)
for $n = 1$ **to** $\max_{i=1,\ldots T} p_i$ **do** `// Main loop`

  for $k = 1$ **to** T, *and* $n \leq p_k$ **do**
    Equations (A.4)
    if $n > 1$ **then for** $j = 1$ **to** $n - 1$ **do**
      Equations (A.5)
    Equations (A.6)
    if $n > 1$ **then for** $j = 1$ **to** $n - 1$ **do**
      Equations (A.7)
    for $t, s = 1$ **to** T *and* $l, m \in \tau_g$
      if $l > n \vee m > n \vee (k = t \wedge n = l) \vee (k = s \wedge n = m)$ **then**
        Equation (A.8a)
      else
        Equation (A.8b)
      Equations (A.9)
    Equations (A.10)

4. Numerical results

We illustrate the method presented in [3] on five examples with $T = 2$ and autocovariances given up to lag 3 (see Table 4) except for a gap at $(1, 2)$ in Examples 1–4 and $(2, 1)$ in Example 5. We have

$$I = \begin{cases} \{(1,0),(1,1),(1,3), \quad (2,0),(2,1),(2,2),(2,3)\} & \text{(Examples 1–4)}, \\ \{(1,0),(1,1),(1,2),(1,3),\ (2,0),(2,1),(2,3)\} & \text{(Example 5)}. \end{cases}$$

The constrained set $E_c(I)$ is the same for all examples:

$$E_c(I) = \{(1,0),(1,1),(1,2),(1,3),\ (2,0),(2,1),(2,2),(2,3)\}.$$

k	0	1	2	3
		Example 1		
$R_1(k)$	1	0.5	?	-0.04892034
$R_2(k)$	1	0.3	-0.09784067	-0.0293522
		Example 2		
$R_1(k)$	1	0.9	?	0.1307306
$R_2(k)$	1	0.3	0.1452563	0.04357688
		Example 3		
$R_1(k)$	1	0.5	?	0.4880678
$R_2(k)$	1	0.3	0.9761356	0.2928407
		Example 4		
$R_1(k)$	1	0.999999	?	0.2995947
$R_2(k)$	1	0.3	0.299595	0.0898785
		Example 5		
$R_1(k)$	1	2	0.7	0.9
$R_2(k)$	1	0.3	?	0.9

Table 1: Autocovariances for the numerical examples. The gaps are designated with question marks.

The examples were constructed by computing autocovariances of periodic autoregressive models specified with their partial autocorrelations. We therefore know the solutions, when they exist, and are able to compare them to the numerical results.

An essential and challenging task in the solution of the ME(K, I) problem is that of finding a systematic method to obtain initial values for the non-linear maximization routines. The method should also be capable of detecting when the problem has no solution (i.e. the sequence K is not a subsequence of any periodically correlated autocovariance sequence). Once an initial value is found, the maximum entropy solution can be found by non-linear maximization using Newton or gradient methods and computing the derivatives of the entropy using the algorithm developed here.

The method developed in [3] does not attack directly the task of finding initial values. It solves a sequence of modified maximum entropy problems constructed so that they have positive definite (p.d.) solutions. We start at step $n = 1$ by filling the gaps with arbitrary numbers, say zeroes, and check if the sequence obtained in this way is p.d. If not, we make it p.d. by adding a sufficiently large constant, c_1, to the lag 0 autocovariances. The resulting sequence is a completion of the sequence $K_{c_1} = \{R_t(k) + c_1\delta_k\}_{(t,k)\in I}$ and therefore can be used as initial value for the modified maximum entropy problem ME(K_{c_1}, I). Then at step $n + 1$ we check if the solution of the ME(K_{c_n}, I) problem is a completion of the original sequence K and, if so, solve our ME(K, I) problem using this solution as initial value (i.e. we set $c_{n+1} = 0$). Otherwise we solve a modified problem ME$(K_{c_{n+1}}, I)$ where $K_{c_{n+1}} = \{R_t(k) + c_{n+1}\delta_k\}_{(t,k)\in I}$ and $c_{n+1} < c_n$ is chosen so that the solution of the ME(K_{c_n}, I) problem can be used as an initial value for the ME$(K_{c_{n+1}}, I)$ problem.

This procedure has excellent properties, see [3]. In a nutshell, $\{c_n\}$ is non-negative monotonically decreasing and can be chosen so that there are three mutually exclusive possibilities which are in one-to-one correspondence to the solvability of the ME(K, I) problem. Namely, (i) a p.d. solution to the ME(K, I) problem exists if and only if c_n becomes equal to 0 for some finite n; (ii) if c_n decreases to 0 without ever reaching it, then only semidefinite solution(s) exist; and (iii) if c_n decreases to a positive limit, then the ME(K, I) problem has no solution.

In the first two examples the maximum entropy p.d. completion exists. The first example illustrates the case when the starting values give at once a p.d. completion, i.e. we can set $c_1 = 0$. In the second example the initial values do not give a p.d. solution and we need to solve a modified problem at the first step. These examples represent typical situations. In the third example, the solution exists but is p.s.d. while the fourth example solution is "almost" p.s.d.. It represents a borderline case. This example is

arguably artificial but it demonstrates vividly the reliability of the method. The sequence c_n decreases quickly and at the fifth iteration we can set $c_5 = 0$. Finally, in the last example there are no solutions.

We did the calculations with the statistical system `R`. We used rather conservative stopping criteria in order to study more reliably the behaviour of the method. In all cases we use zeroes as initial values for the gaps.

Example 1. Here the gap is at $(1, 2)$, i.e. $t = 1$, $k = 2$. The "true" maximum entropy solution is obtained for $R_1(2) = 0.15$. Table 2 gives results obtained with our procedure for this example. The first column gives the values tried for c_n. The results for each iteration at the corresponding values of c_n are given in the following columns, starting with the iteration number, then the value for the missing autocovariance coefficient, the entropy, and the value of the gradient. Here it is possible to set immediately $c_n = 0$ and after 3 iterations the procedure leads to the expected solution, $R_1(2) = 0.15$.

c_n	i	$R_1(2)$	entropy	gradient
0	0	0.0000	-0.2660	0.3814
	1	0.1355	-0.2384	0.0349
	2	0.1499	-0.2382	3.3148e-05
	3	0.1500	-0.2382	2.8209e-14

Table 2: Results for Example 1. The initial value $R_1(2) = 0$ is admissible, so $c_1 = 0$ and the optimal solution $R_1(2) = 0.15$ is reached quickly.

Example 2. In this example we again have a p.d. solution but the value zero is not admissible for c_1. The results are given in Table 3. The optimal value, 0.27, of $R_1(2)$ was reached after two iterations (with $c_1 = 1$ and $c_2 = 0$, respectively) of step 3 of our method. The ME completion here is relatively far from the p.s.d. border, as measured, for example, by the absolute value of the largest partial autocorrelation coefficient. This is equal to 0.9, not very close to 1. So, the example shows also that a non-zero c_1 may be needed even if the problem is not close to singularity.

Example 3. This example is similar to Example 1 but it is set up so that the solution is p.s.d.. The value 0.15 is an optimal solution for the missing $R_1(0)$, as in Example 1. For this singular example the procedure leads to a sequence of c_n values that monotonically decreases to 0. Table 4 gives the

c_n	i	$R_1(2)$	entropy	gradient
1	0	0.0000	0.5623	0.0679
	1	0.1619	0.5679	0.00146
	2	0.1655	0.5679	1.1178e-08
	3	0.1655	0.5679	-7.2435e-16
0	0	0.1655	-1.1520	5.4523
	1	0.2154	-0.9776	2.0404
	2	0.2604	-0.9262	0.3238
	3	0.2699	-0.9247	0.0018
	4	0.2700	-0.9247	3.7871e-10
	5	0.2700	-0.9247	3.4084e-10

Table 3: Results for Example 2. The initial value $R_1(2) = 0$ is NOT admissible. However the optimal value, $R_1(2) = 0.16552$, obtained for the modified problem with $c_1 = 1$ is admissible for the original problem, so $c_2 = 0$. The optimal solution here is $R_1(2) = 0.27$.

results obtained for the first eight values of c_n as well as for its 80th value. For each value of c_n, we give only the results for the last iteration of the Newton-Raphson's algorithm. Table 4 is in agreement with the theory which says that $c_n \to 0$ in the p.s.d. case.

Example 4. In this example the solution is "almost" p.s.d.. It would have been exactly p.s.d. if $R_1(1) = 1$, a difference of only 0.000001 from the value used in this example, see Table 4. The optimal solution for $R_1(2)$ is 0.2999997. Table 5 gives results obtained for this example. Here the procedure is slower than in Examples 1–2 but, bearing in mind the close proximity to singularity, the fact that the solution was obtained with only 5 iterations of step 3 of the method seems an excellent result.

Example 5. Our last example illustrates the case when the given sequence is not completable. Here this is straightforward to check by observing that all correlations corresponding to a proper autocovariance sequence must be in the interval $[-1, 1]$, while the given autocovariances imply that the correlation coefficient $R(3,2)/\sqrt{R(3,3)R(2,2)} = R_1(1)/\sqrt{R_1(0)R_2(0)} = 2 > 1$. Table 6 gives the results obtained for the first eight values of c_n as well as for its 83th value. For each values of c_n, as in the previous example, we give only the results for the last iteration of the Newton-Raphson's algorithm. The procedure converges quickly to the value $c = 1.069440 > 0$, which shows that

c_n	i	$R_1(2)$	entropy	gradient
1.000000	3	0.1362538	0.5102	-8.2608e-09
0.050000	3	0.1616193	-1.2162	-2.2817e-10
0.002500	3	0.1511813	-2.7275	3.7530e-10
0.000125	2	0.1500620	-4.2261	4.6311e-12
6.250e-06	2	0.1500031	-5.7240	-1.5291e-13
3.125e-07	2	0.1500002	-7.2219	-6.4537e-07
1.563e-08	1	0.1500000	-8.7197	-5.8958e-08
7.813e-10	1	0.1500000	-10.2176	-9.5701e-09
...	...	...	...	...
7.994e-16	2	0.1500000	-17.1731	-0.0012

Table 4: Results for Example 3. The solution here is p.s.d.. In this case c_n converges to 0 but can never be set equal to 0. In floating point calculations this cannot be detected exactly but the decision rules used for our examples stopped the algorithm after solving the 80th modified problem (with $c_{80} = 7.994 \cdot 10^{-16}$) and declared that the solution is p.s.d..

no solution exists.

The gradients are large for the same reasons as in the p.s.d. case. In fact, when c_n converges to $c = 1.069440 > 0$, $R_1(2)$ converges to 0.4410555, which provides a p.s.d. completion to the modified problem $\mathrm{ME}(K_c, I)$ for the modified sequence $K_c = \{R_t(k) + c\}_{(t,k)\in I}$, not to the original problem, $\mathrm{ME}(K, I)$, see [3] for further details.

5. Remarks on p.s.d. solutions

From numerical viewpoint there is little distinction between the "exact" and "almost" p.s.d. case. One might expect erratic behaviour of numerical algorithms in these cases. It is therefore interesting that our method not only successfully deals with them but it also distinguishes the two cases.

For example, in the singular case illustrated by Example 3 the solution changes very little from about c_{15}, but it does change. On the other hand, the "almost" p.s.d. case of Example 4 reached the value $c_n = 0$ for $n = 5$ iterations and finished successfully. We run Examples 3–5, and many other similar examples, for far more iterations that would be needed in practice in order to get a better understanding of the behaviour of the method. Our results were qualitatively similar—the method finishes successfully and fails to distinguish "almost" from "exact" p.s.d. only when the solution is extremely

c_n	i	$R_1(2)$	entropy	gradient
1.000000	0	0.0000000	0.5184	0.1046
	1	0.2152802	0.5300	0.0045
	2	0.2253450	0.5300	3.5921e-07
	3	0.2253458	0.5300	-6.5702e-13
0.050000	0	0.2253458	-1.6121	18.3505
	1	0.2452564	-1.3671	8.2140
	2	0.2738726	-1.2187	2.8691
	3	0.2960202	-1.1841	0.3400
	4	0.2992355	-1.1836	0.0007
	5	0.2992426	-1.1836	-1.7116e-12
0.002500	0	0.2992426	-2.7068	30.9271
	1	0.2999629	-2.6953	1.3467
	2	0.2999965	-2.6953	0.0001
	3	0.2999965	-2.6953	-2.1290e-12
0.000125	0	0.2999965	-4.1905	48.7861
	1	0.2999996	-4.1904	0.0147
	2	0.2999996	-4.1904	5.2170e-10
0.000000	0	0.2999996	-6.6560	16725.18
	1	0.2999997	-6.6554	33.9126

Table 5: Results for Example 4. The solution here is almost p.s.d.. Despite that the algorithm finds an admissible solution after solving 4 modified problems and after only one additional iteration gives the optimal value, $R_1(2) = 0.2999997$. The large gradients are discussed in the main text and Appendix B.

c_n	i	$R_1(2)$	entropy	gradient
2.000000	4	0.4514833	0.7068	-7.8642e-08
1.073418	4	0.4477606	-2.9850	-1.9047e-05
1.070620	8	0.4431905	-4.1350	-0.3180
1.069847	6	0.4417908	-5.1751	-4.8700e-05
1.069569	8	0.4412777	-6.2893	-9.9963e-05
1.069480	7	0.4411105	-7.3921	-12.6808
1.069451	6	0.4410558	-8.4504	-14.0478
1.069440	1	0.4410556	-11.1154	-1077458
...	...	...	...	...
1.069440	1	0.4410555	-20.8842	-3.3077e+14

Table 6: Results for Example 5. The sequence c_n converges to a positive value since the maximum entropy problem has no solution here.

close to being p.s.d.. Note that the "distance" of 10^{-6} from singularity in Example 4 is not extremely close in this context. In some examples it seemed not possible to make the gradient at the declared solution sufficiently small. However, the huge value of the gradient in examples like Example 5 is not surprising since as we are approaching a p.s.d. solution (of a possibly modified problem, see end of Section 4) the entropy diverges to $-\infty$, see Appendix Appendix B. It may be better to maximize e^h in such cases and we did some limited experiments but the results are similar. So, we may confidently conclude that our method is very reliable even in cases close to singularity. Since it also detects when the completion problem has no solution, it is a complete all round method.

In the p.s.d. case there are, in general, infinitely many solutions. The one chosen by the algorithm discussed here is a limit of solutions of slightly modified p.d. problems, which seems a desirable property. In many cases it also maximizes $\sum_t \log \sigma_t^2$, where the sum is over t such that $\sigma_t \neq 0$, which seems a desirable property, as well, but we do not know if this is always the case.

A p.s.d. solution implies that observations for one or more seasons are exact linear combinations of past values. It is possible to interpret a p.s.d. solution as corresponding to unit root or periodic integration models (see [7, Chapter 4]), especially in applications to economic data.

The following example illuminates some of the points discussed above. Let

$T = 2$, $R_1(0) = R_2(0) = 1$, $R_1(2) = \alpha$, $R_2(4) = \beta$, where $|\alpha| < 1$ and $|\beta| < 1$. Here $E_c(I) = \{(1,0),(1,1),(1,2),(1,3),(1,4),(2,0),(2,1),(2,2),(2,3)\}$. So, the gaps are $(1,1),(1,2),(1,3),(2,1),(2,3)$. For the maximum entropy solution we have $R_1(1) = R_1(2) = R_1(3) = R_2(1) = R_2(3) = 0$ which corresponds to the PAR(4,2) model

$$X_t = \begin{cases} \alpha X_{t-2} + \epsilon_t & \text{for odd } t, \\ \beta X_{t-4} + \epsilon_t & \text{for even } t, \end{cases}$$

where ϵ_t is periodic white noise. Now, let $\beta = 1$, i.e. $R_2(4) = 1$. In this case the solution is p.s.d. The algorithm will try modified problems with $R_1(0) = R_2(0) = 1 + c$. For each $c > 0$ the solution of the modified problem is

$$\begin{gathered} R_1(0) = R_2(0) = 1 + c, \qquad R_1(2) = \alpha, \qquad R_2(4) = \beta = 1, \\ R_1(1) = R_1(2) = R_1(3) = R_2(1) = R_2(3) = 0, \end{gathered} \tag{4}$$

which corresponds to the PAR(4,2) model

$$X_t = \begin{cases} \alpha_c X_{t-2} + \epsilon_t & \text{for odd } t, \\ \beta_c X_{t-4} + \epsilon_t & \text{for even } t, \end{cases} \tag{5}$$

where $\alpha_c = \alpha/(1+c)$, $\beta_c = 1/(1+c)$, and ϵ_t is periodic white noise.

As $c \to 0$ we obtain in the limit $R_1(0) = R_2(0) = 1$, the remaining autocovariances being as in Equation (4). Since $R_2(4) = 1$ we can see that if $\{X_t\}$ is a process with these autocovariances, then with probability one $X_t = X_{t-4}$ for even t and so the values of the time series for even t represent a periodic function. On the other hand, the limit of the models (5) is

$$X_t = \begin{cases} \alpha X_{t-2} + \epsilon_t & \text{for odd } t, \\ X_{t-4} + \epsilon_t & \text{for even } t, \end{cases}$$

where $\sigma_t^2 = \mathrm{Var}(\epsilon_t) = 0$ for even t. In applications, especially to economic data, it may be more natural to consider the above model with $\sigma_{2t}^2 > 0$, which corresponds to a periodic integration model. Here we have essentially separate models for the two seasons. This of course does not hold for more general autocorrelation patterns.

Appendix A. PLD algorithm steps

Appendix A.1. Initialisation step

For $k = 1, \ldots, T,\ t,\ \ s = 1, \ldots, T,\ l, m \in \tau_g$, set the following initial values.

$$\beta_k(0) = \sigma_k^{f2}(0) = \sigma_k^{b2}(0) = R_k(0), \tag{A.1}$$

$$\frac{\partial \sigma_k^{f2}(0)}{\partial R_t(l)} = \frac{\partial \sigma_k^{b2}(0)}{\partial R_t(l)} = 0, \tag{A.2}$$

$$\frac{\partial^2 \sigma_k^{f2}(0)}{\partial R_t(l) \partial R_s(m)} = \frac{\partial^2 \sigma_k^{b2}(0)}{\partial R_t(l) \partial R_s(m)} = 0. \tag{A.3}$$

Appendix A.2. PLD step for the parameters

These are the standard PLD calculations, see [13].

The equations for step n are:

$$A_k(n) = R_k(n) + \sum_{j=1}^{n-1} a_k^f(n-1, j) R_{(k-j)[T]}(n-j), \tag{A.4a}$$

$$\beta_k(n) = \frac{A_k(n)}{\sigma_k^f(n-1)\sigma_k^b(n-1)}, \tag{A.4b}$$

$$\sigma_k^{f2}(n) = \sigma_k^{f2}(n-1) - \frac{A_k(n)^2}{\sigma_{k-1}^{b2}(n-1)}, \tag{A.4c}$$

$$\sigma_k^{b2}(n) = \sigma_{k-1}^{b2}(n-1) - \frac{A_k(n)^2}{\sigma_k^{f2}(n-1)}, \tag{A.4d}$$

$$a_k^f(n, n) = - - \frac{A_k(n)}{\sigma_{k-1}^{b2}(n-1)}, \tag{A.4e}$$

$$a_k^b(n, n) = - \frac{A_k(n)}{\sigma_k^{f2}(n-1)}. \tag{A.4f}$$

The following calculations are done after the above when $n > 1$.

$$a_k^f(n, j) = a_k^f(n-1, j) + a_k^f(n, n) a_{k-1}^b(n-1, n-j), \tag{A.5a}$$

$$a_k^b(n, j) = a_{k-1}^b(n-1, j) + a_k^b(n, n) a_k^f(n-1, n-j), \tag{A.5b}$$

(for $j = 1, \ldots, n-1$).

Appendix A.3. Gradient PLD step

For $t = 1, \ldots, T,\ l \in \tau_g$,

$$\frac{\partial A_k(n)}{\partial R_t(l)} = \begin{cases} 0, & \text{when } l > n, \\ 1, & \text{when } k = t \text{ and } n = l, \\ \delta_{t,(k-n+l)[T]} a_k^f(n-1, n-l) & \\ \quad + \sum_{j=1}^{n-1} \dfrac{\partial a_k^f(n-1,j)}{\partial R_t(l)} R_{(k-j)[T]}(n-j), & \text{otherwise.} \end{cases} \tag{A.6a}$$

$$\frac{\partial \sigma_k^{f2}(n)}{\partial R_t(l)} = \frac{\partial \sigma_k^{f2}(n-1)}{\partial R_t(l)} - \frac{2A_k(n)\frac{\partial A_k(n)}{\partial R_t(l)}}{\sigma_{k-1}^{b2}(n-1)} + \frac{A_k(n)^2 \frac{\partial \sigma_{k-1}^{b2}(n-1)}{\partial R_t(l)}}{(\sigma_{k-1}^{b2}(n-1))^2}, \tag{A.6b}$$

$$\frac{\partial \sigma_k^{b2}(n)}{\partial R_t(l)} = \frac{\partial \sigma_{k-1}^{b2}(n-1)}{\partial R_t(l)} - \frac{2A_k(n)\frac{\partial A_k(n)}{\partial R_t(l)}}{\sigma_k^{f2}(n-1)} + \frac{A_k(n)^2 \frac{\partial \sigma_k^{f2}(n-1)}{\partial R_t(l)}}{(\sigma_k^{f2}(n-1))^2}, \tag{A.6c}$$

$$\frac{\partial a_k^f(n,n)}{\partial R_t(l)} = -\frac{\frac{\partial A_k(n)}{\partial R_t(l)}}{\sigma_{k-1}^{b2}(n-1)} + \frac{A_k(n)\frac{\partial \sigma_{k-1}^{b2}(n-1)}{\partial R_t(l)}}{(\sigma_{k-1}^{b2}(n-1))^2}, \tag{A.6d}$$

$$\frac{\partial a_k^b(n,n)}{\partial R_t(l)} = -\frac{\frac{\partial A_k(n)}{\partial R_t(l)}}{\sigma_k^{f2}(n-1)} + \frac{A_k(n)\frac{\partial \sigma_k^{f2}(n-1)}{\partial R_t(l)}}{(\sigma_k^{f2}(n-1))^2}. \tag{A.6e}$$

And, if $n > 1$, for $j = 1, \ldots, n-1$,

$$\begin{aligned} \frac{\partial a_k^f(n,j)}{\partial R_t(l)} = \frac{\partial a_k^f(n-1,j)}{\partial R_t(l)} &+ \frac{\partial a_k^f(n,n)}{\partial R_t(l)} a_{k-1}^b(n-1, n-j) \\ &+ a_k^f(n,n) \frac{\partial a_{k-1}^b(n-1,n-j)}{\partial R_t(l)}, \end{aligned} \tag{A.7a}$$

$$\begin{aligned} \frac{\partial a_k^b(n,j)}{\partial R_t(l)} = \frac{\partial a_{k-1}^b(n-1,j)}{\partial R_t(l)} &+ \frac{\partial a_k^b(n,n)}{\partial R_t(l)} a_k^f(n-1, n-j) \\ &+ a_k^b(n,n) \frac{\partial a_k^f(n-1,n-j)}{\partial R_t(l)}. \end{aligned} \tag{A.7b}$$

Appendix A.4. Hessian PLD step

For $t, s = 1, \ldots, T,\ l, m \in \tau_g$,

If $l > n$ or $m > n$ or ($k = t$ and $n = l$) or ($k = s$ and $n = m$), put

$$\frac{\partial^2 A_k(n)}{\partial R_t(l)\partial R_s(m)} = 0. \tag{A.8a}$$

Otherwise put

$$\frac{\partial^2 A_k(n)}{\partial R_t(l)\partial R_s(m)} = \delta_{t,(k-n+l)[T]} \frac{\partial a_k^f(n-1, n-l)}{\partial R_s(m)} + \delta_{s,(k-n+m)[T]} \frac{\partial a_k^f(n-1, n-m)}{\partial R_t(l)} + \sum_{j=1}^{n-1} \frac{\partial^2 a_k^f(n-1, j)}{\partial R_t(l)\partial R_s(m)} R_{(k-j)[T]}(n-j), \tag{A.8b}$$

$$\frac{\partial^2 a_k^f(n,n)}{\partial R_t(l)\partial R_s(m)} = \frac{\frac{\partial \sigma_{k-1}^{b2}(n-1)}{\partial R_s(m)} \frac{\partial A_k(n)}{\partial R_t(l)}}{(\sigma_{k-1}^{b2}(n-1))^2} - \frac{\frac{\partial^2 A_k(n)}{\partial R_t(l)\partial R_s(m)}}{\sigma_{k-1}^{b2}(n-1)} + \frac{\frac{\partial A_k(n)}{\partial R_s(m)} \frac{\partial \sigma_{k-1}^{b2}(n-1)}{\partial R_t(l)}}{(\sigma_{k-1}^{b2}(n-1))^2} - 2\frac{A_k(n) \frac{\partial \sigma_{k-1}^{b2}(n-1)}{\partial R_s(m)} \frac{\partial \sigma_{k-1}^{b2}(n-1)}{\partial R_t(l)}}{(\sigma_{k-1}^{b2}(n-1))^3} + \frac{A_k(n) \frac{\partial^2 \sigma_{k-1}^{b2}(n-1)}{\partial R_t(l)\partial R_s(m)}}{(\sigma_{k-1}^{b2}(n-1))^2}, \tag{A.9a}$$

$$\frac{\partial^2 a_k^b(n,n)}{\partial R_t(l)\partial R_s(m)} = \frac{\frac{\partial \sigma_k^{f2}(n-1)}{\partial R_s(m)} \frac{\partial A_k(n)}{\partial R_t(l)}}{(\sigma_k^{f2}(n-1))^2} - \frac{\frac{\partial^2 A_k(n)}{\partial R_t(l)\partial R_s(m)}}{\sigma_k^{f2}(n-1)} + \frac{\frac{\partial A_k(n)}{\partial R_s(m)} \frac{\partial \sigma_k^{f2}(n-1)}{\partial R_t(l)}}{(\sigma_k^{f2}(n-1))^2} - 2\frac{A_k(n) \frac{\partial \sigma_k^{f2}(n-1)}{\partial R_s(m)} \frac{\partial \sigma_k^{f2}(n-1)}{\partial R_t(l)}}{(\sigma_k^{f2}(n-1))^3} + \frac{A_k(n) \frac{\partial^2 \sigma_k^{f2}(n-1)}{\partial R_t(l)\partial R_s(m)}}{(\sigma_k^{f2}(n-1))^2}, \tag{A.9b}$$

$$\frac{\partial^2 \sigma_k^{f2}(n)}{\partial R_t(l)\partial R_s(m)} = \frac{\partial^2 \sigma_k^{f2}(n-1)}{\partial R_t(l)\partial R_s(m)} - 2\frac{\frac{\partial A_k(n)}{\partial R_s(m)} \frac{\partial A_k(n)}{\partial R_t(l)}}{\sigma_{k-1}^{b2}(n-1)} - 2\frac{A_k(n) \frac{\partial^2 A_k(n)}{\partial R_t(l)\partial R_s(m)}}{\sigma_{k-1}^{b2}(n-1)} - 2\frac{A_k(n) \frac{\partial \sigma_{k-1}^{b2}(n-1)}{\partial R_t(l)}}{(\sigma_{k-1}^{b2}(n-1))^3} + 2\frac{A_k(n) \frac{\partial A_k(n)}{\partial R_t(l)} \frac{\partial \sigma_{k-1}^{b2}(n-1)}{\partial R_s(m)}}{(\sigma_{k-1}^{b2}(n-1))^2} + 2\frac{A_k(n) \frac{\partial A_k(n)}{\partial R_s(m)} \frac{\partial \sigma_{k-1}^{b2}(n-1)}{\partial R_t(l)}}{(\sigma_{k-1}^{b2}(n-1))^2} + \frac{A_k(n)^2 \frac{\partial^2 \sigma_{k-1}^{b2}(n-1v)}{\partial R_t(l)\partial R_s(m)}}{(\sigma_{k-1}^{b2}(n-1))^2}, \tag{A.9c}$$

$$\frac{\partial^2\sigma_k^{b2}(n)}{\partial R_t(l)\partial R_s(m)} = \frac{\partial^2\sigma_{k-1}^{b2}(n-1)}{\partial R_t(l)\partial R_s(m)} - 2\frac{\frac{\partial A_k(n)}{\partial R_s(m)}\frac{\partial A_k(n)}{\partial R_t(l)}}{\sigma_k^{f2}(n-1)} - 2\frac{A_k(n)\frac{\partial^2 A_k(n)}{\partial R_t(l)\partial R_s(m)}}{\sigma_k^{f2}(n-1)}$$
$$- 2\frac{A_k(n)\frac{\partial\sigma_k^{f2}(n-1)}{\partial R_t(l)}}{(\sigma_k^{f2}(n-1))^3} + 2\frac{A_k(n)\frac{\partial A_k(n)}{\partial R_t(l)}\frac{\partial\sigma_k^{f2}(n-1)}{\partial R_s(m)}}{(\sigma_k^{f2}(n-1))^2}$$
$$+ 2\frac{A_k(n)\frac{\partial A_k(n)}{\partial R_s(m)}\frac{\partial\sigma_k^{f2}(n-1)}{\partial R_t(l)}}{(\sigma_k^{f2}(n-1))^2} + \frac{A_k(n)^2\frac{\partial^2\sigma_k^{f2}(n-1)}{\partial R_t(l)\partial R_s(m)}}{(\sigma_k^{f2}(n-1))^2}. \quad \text{(A.9d)}$$

And, if $n > 1$, for $j = 1, \ldots, n-1$,

$$\frac{\partial^2 a_k^f(n,j)}{\partial R_t(l)\partial R_s(m)} = \frac{\partial^2 a_k^f(n-1,j)}{\partial R_t(l)\partial R_s(m)} + a_k^f(n,n)\frac{\partial^2 a_{k-1}^b(n-1,n-j)}{\partial R_t(l)\partial R_s(m)}$$
$$+a_{k-1}^b(n,n-j)\frac{\partial^2 a_k^f(n,n)}{\partial R_t(l)\partial R_s(m)} + \frac{\partial a_k^f(n,n)}{\partial R_t(l)}\frac{\partial a_{k-1}^b(n-1,n-j)}{\partial R_s(m)}$$
$$+\frac{\partial a_k^f(n,n)}{\partial R_s(m)}\frac{\partial a_{k-1}^b(n-1,n-j)}{\partial R_t(l)}, \quad \text{(A.10a)}$$

$$\frac{\partial^2 a_k^b(n,j)}{\partial R_t(l)\partial R_s(m)} = \frac{\partial^2 a_{k-1}^b(n-1,j)}{\partial R_t(l)\partial R_s(m)} + a_k^b(n,n)\frac{\partial^2 a_k^f(n-1,n-j)}{\partial R_t(l)\partial R_s(m)}$$
$$+a_k^f(n,n-j)\frac{\partial^2 a_k^b(n,n)}{\partial R_t(l)\partial R_s(m)} + \frac{\partial a_k^b(n,n)}{\partial R_t(l)}\frac{\partial a_k^f(n-1,n-j)}{\partial R_s(m)}$$
$$+\frac{\partial a_k^b(n,n)}{\partial R_s(m)}\frac{\partial a_k^f(n-1,n-j)}{\partial R_t(l)}. \quad \text{(A.10b)}$$

Appendix B. Gradient of e^{entropy}

Let h be the entropy rate and consider e^h. Maximizing h and e^h is equivalent. Let y be any of the parameters w.r.t. which we are maximizing. We have $\frac{\partial}{\partial y}e^h = e^h\frac{\partial h}{\partial y}$. Hence, $\frac{\partial h}{\partial y} = e^{-h}\frac{\partial}{\partial y}e^h$. Therefore the gradient of h is e^{-h} times the gradient of e^h. Near the optimal values both partial derivatives are "close" to zero but differ by a factor of e^{-h}. So, near the p.s.d. border where h goes to $-\infty$ and e^{-h} goes to ∞, the derivative of H can be large even very close to the optimal value.

Acknowledgements

Part of this work was supported by the Interuniversity Attraction Pole (IAP) research network in Statistics P5/24. The first author gratefully acknowledges the support of LJK-IMAG at Joseph Fourier University in Grenoble, France, where part of this work was completed during his visit in February–March 2007. We thank the referees and the associate editor for their constructive comments.

References

[1] D. Alpay, A. Chevreuil, and P. Loubaton. An extension problem for discrete-time periodically correlated stochastic processes. *J. Time Ser. Anal.*, 22(1):1–11, 2001.

[2] G. Boshnakov and S. Lambert-Lacroix. *pcme: Maximum entropy estimation of periodically correlated time series*, 2009. R package version 0.51 or later, available from `http://www.maths.manchester.ac.uk/~gb/Rpackages/`.

[3] G. N. Boshnakov and S. Lambert-Lacroix. Maximum entropy for periodically correlated processes from nonconsecutive autocovariance coefficients. *J. Time Series Anal.*, 30(5):467–486, 2009.

[4] J. Burg. *Maximum entropy spectral analysis.* PhD thesis, Department of Geophysics, Stanford University, Stanford, CA, 1975.

[5] J. P. Burg. Maximum entropy spectral analysis. In *Proceedings of 37th Meeting of Society of Exploration Geophysicists*, pages 34–41, 1967.

[6] G. Castro and V. Girardin. Maximum of entropy and extension of covariance matrices for periodically correlated and multivariate processes. *Stat. Probab. Lett.*, 59(1):37–52, 2002.

[7] P. H. Franses and R. Paap. *Periodic time series models.* Advanced Texts in Econometrics. Oxford: Oxford University Press, 2004.

[8] E. Gladishev. Periodically correlated random sequences. *Soviet Math.*, 2:385–388, 1961.

[9] I. Hindrayanto, S. J. Koopman, and M. Ooms. Exact maximum likelihood estimation for non-stationary periodic time series models. *Computational Statistics & Data Analysis*, 54(11):2641 – 2654, 2010.

[10] K. W. Hipel and A. I. McLeod. *Time series modelling of water resources and environmental systems*. Developments in water science; 45. London; Amsterdam: Elsevier, 1994.

[11] H. L. Hurd and A. Miamee. *Periodically correlated random sequences. Spectral theory and practice.* Wiley Series in Probability and Statistics; Wiley Interscience. Hoboken, NJ: John Wiley & Sons, 2007.

[12] B. M. Iqelan. *Periodically correlated time series: models and examples*. PhD thesis, The University of Manchester, 2007.

[13] S. Lambert-Lacroix. Extension of autocovariance coefficients sequence for periodically correlated processes. *Journal of Time Series Analysis*, 26(6):423–435, 2005.

[14] D. N. Politis. Moving average processes and maximum entropy. *IEEE Trans. Inf. Theory*, 38(3):1174–1177, 1992.

[15] N. Rozario and A. Papoulis. Spectral estimation from nonconsecutive data. *IEEE Trans. Inf. Theory*, 33:889–894, 1987.

[16] H. Sakai. Circular lattice filtering using pagano's method. *IEEE Trans. Acoust., Speech Signal Process*, 30:279–87, 1982.

[17] E. Serpedin, F. Panduru, I. Sari, and G. B. Giannakis. Bibliography on cyclostationarity. *Signal Process.*, 85(12):2233–2303, 2005.