Periodic Autoregressive Time Series Models in R: The partsm Package.

Javier López-de-Lacalle *
Universidad del País Vasco
javlacalle@yahoo.es

Abstract

This introduction to the R package **partsm** is a (slightly) modified version of López-de Lacalle (2005).

It is well-known that some of the macroeconomic time series display stochastic trends, moreover, when working with seasonally observed data stochastic seasonal cycles may exist as well. When these components, trend and seasonality, do not evolve independently, traditional differencing filters may not be suitable. According to periodic autoregressive time series models, a seasonally varying autoregressive parameters and a periodic differencing filter are proposed for that case.

This paper focuses on practical issues showing the use of the **partsm** R-package. This package allows the user to check for periodicity in the data, fit a periodic autoregressive model of order p, PAR(p), select the periodic autoregressive lag order parameter, test for periodic integration, fit a periodically integrated autoregressive model up to order 2, PIAR, as well as to perform out-of-sample forecasts.

Keywords: Time series, PAR models, periodic integration, R.

1. Introduction

It is well-known that some of the macroeconomic time series display stochastic trends, moreover, when working with seasonally observed data stochastic seasonal cycles may exist as well. When these components, trend and seasonality, do not evolve independently, traditional differencing filters may not

*I acknowledge Matthieu Stigler's contribution for the upgrade and maintenance of the package.

be suitable. According to periodic autoregressive time series models, a seasonally varying autoregressive parameters and a periodic differencing filter are proposed for that case. For a review of this literature see Franses (1996), Franses and Paap (2004), and references therein.

This paper focuses on practical issues showing the use of the **partsm** R-package. This package allows the user to check for periodicity in the data, fit a periodic autoregressive model of order p, PAR(p), select the periodic autoregressive lag order parameter, test for periodic integration, fit a periodically integrated autoregressive model up to order 2, PIAR, as well as to perform out-of-sample forecasts.

The remaining of the paper is organized as follows. Section 2 briefly reviews the statistical issues the **partsm** R-package is concerned, namely periodic autoregressive models and periodic integration. Section 3 describes the package. Section 4 puts into practice the tools implemented in the package showing how to use them and interpreting the results accordingly.

2. Theoretical overview

This section reviews the main theoretical concerns entailed in the process of fitting periodic models and testing for a unit root in PAR models. To have a further insight into this models see Franses (1996), Franses and Paap (2004), and references therein. PAR models are intended for seasonally observed data, particularly quarterly and monthly data. To save space, hereafter we will consider quarterly data, $S = 4$.

2.1. Notation and representation of PAR models

The *univariate representation* of a PAR(p) model is as follows,

$$y_t = \phi_{1s} y_{t-1} + ... + \phi_{ps} y_{t-p} + \epsilon_t, \quad \epsilon_t \sim iid(0,1), \tag{1}$$

for $s = 1, ..., 4$, for $t = 1, 2, ..., n$, where n is the number of observations. Hence, the autoregressive parameters vary with the season for each lag.

Since a PAR(p) entails four different AR(p) models, one for each season, it is useful to rewrite (1) as the *multivariate representation* or *vector of quarters representation.*

$$\Phi_0 Y_{s,T} = \Phi_1 Y_{s,T-1} + ... + \Phi_P Y_{s,T-P} + \epsilon_T, \quad \epsilon_T \sim iid(0,1), \tag{2}$$

where $\Phi_0, \Phi_1, ..., \Phi_P$ are (4×4) parameter matrices with the parameters in (1) as follows:

$$\begin{aligned} \Phi_0(i,j) &= 1 \quad if \quad i = j \\ &= 0 \quad if \quad j > i \end{aligned}$$

$$\begin{aligned} &= -\phi_{i-j,i} \quad if \quad j < y \\ \Phi_k\,(i,j) &= \phi_{i+4k-j,i}, \end{aligned}$$

for $i, j = 1, 2, 3, 4$ and $k = 1, 2, ..., P$. The univariate model of order p turns into a multivariate model of order $P = 1+[(p-1)/4]$, where $[x]$ is the integer part of x.

Notice that in (1) each lag leads to previous observations in the seasonally observed data, whereas in (2) lags have effect on the annually observed data in each season. That is, y_{t-1} is the observation immediately previous to y_t, and $Y_{s,T-1}$ is the observation in season s previous to the year T.

Finally, if the roots of the factorized AR(p) model related to each season are all real values, the PAR(p) model can be represented as

$$\begin{aligned} y_t - \alpha_s\, y_{t-1} &= \beta_{1s}\,(y_{t-1} - \alpha_{s-1}\, y_{t-2}) + ... + \\ &+ \beta_{(p-1)s}\,(y_{t-(p-1)} - \alpha_{s-(p-1)}\, y_{t-p}) + \epsilon_t, \quad \epsilon_t \sim iid(0,1). \end{aligned} \tag{3}$$

called the *periodically differenced form of* (1).

2.2. Periodic integration

Basically, a time series with a unit root, y_t, is periodically integrated if there exist some α_s for $s = 1, 2, ..., 4$, in such a way that the transformed series $(1-\alpha_s\, B)\, y_t$ does not contain a unit root, where B is the backward operator. The definition of Franses (1996) is transcribed below.

> A quarterly time series y_t is said to be *periodically integrated of order 1* [PI] when the differencing filter $(1 - \alpha_s\, B)$ is needed to remove the stochastic trend from y_t, where α_s are seasonally varying parameters with the property that $\alpha_1\alpha_2\alpha_3\alpha_4 = 1$ and $\alpha_s \neq \alpha$ for all $s = 1, 2, 3, 4$.

At present, the package **partsm** only allow to estimate periodically integrated autoregressive models up to order 2[1]. Taking the periodically differenced representation, the following model can be estimated by non-linear least squares.

$$y_t - \alpha_s y_{t-1} = \beta_s(y_{t-1} - \alpha_{s-1} y_{t-2}) + \epsilon_t, \quad \epsilon_t \sim iid(0,1), \tag{4}$$

under the non-linear restriction $\prod_{s=1}^{4} \alpha_s = 1$ for $s = 1, ..., 4$. Obviously, for a first order PIAR process β parameters are equal to zero. This model can be estimated by non-linear least squares.

[1]PAR models use to be more parsimonious models and a first or second order model may be suitable for the data. In Franses (1996), only one out of the eleven series analysed required a higher order PAR model.

Note that the restriction above is fulfilled in these particular cases, among others. When $\alpha_s = 1$ for $s = 1, 2, 3, 4$, and $\alpha_s = -1$ for $s = 1, 2, 3, 4$. The former case give rise to the $(1 - L)$ differencing filter, whereas the latter entails the $(1 + L)$ differencing filter. Therefore, the restriction may involve the long run unit root 1, or the seasonal unit root -1, or neither of them.

When the hypothesis above cannot be rejected it is said that the process is a *PAR process for a I(1) times series*, PARI. Otherwise, the PAR model is known as a *periodically integrated AR model*, PIAR, and the periodic differencing filter is obtained from the α_s estimates in equation (4).

In Section 4 we will see how to carry out this analysis following a common strategy for the empirical analysis. First, we will take a look at the package we will use for it.

3. The partsm package

3.1. Description

This section documents the **partsm** package version 1.0 built on the R language and environment for statistical computing and graphics (Chambers, 1998; R Development Core Team, 2011). The package performs some of the relevant tests and models for fitting periodic autoregressive time series model introduced in the previous section.

The package is distributed under the General Public License [GPL] version 2 or newer. The terms of this license are in a file called COPYING which you should receive with R. After reading the terms of the license, the user will understand that the datasets and software are provided in good faith, but the author does not warrant their accuracy nor can be held responsible for the consequences of their use.

The source code and binaries of the package are available at CRAN (`http://www.cran.r-project.org/`). To add it as a package copy the binaries in the subdirectory 'library' where R is installed[2]. Alternatively, download the package source and install it with `R CMD INSTALL partsm_1.0.tar.gz`. To install it from an R-console type `install.packages("partsm")` and select a mirror near to your location.

3.2. Classes and methods

To store the relevant information provided by the implemented functions, the following classes are defined: `fit.partsm` and `fit.piartsm` contain the information from a fitted AR, PAR, or PIAR model; `Ftest.partsm` and

[2]Type `R.home()` in an R-console to find out the home directory.

`LRur.partsm` store the information from the statistical tests in the package; and `pred.piartsm` contains information on a PIAR model forecasts. For more information, see the standard help pages of the package[3]

Likewise, some methods are defined for objects of the classes cited above. To display the information in each object, the `show` method displays the main results in a friendly format, whereas `summary` extends the information provided by `show`. To build the matrices for the multivariate representation, the `PAR.MVrepr` method can be applied on objects of class `fit.partsm` or `fit.piartsm`.

4. Examples and applications

This Section complements the standard help pages provided with the package carrying out an entire application. The use and interpretation of the functions implemented in the package is described following the same steps as in a real application, devoting a subsection for each one of them.

It is worth describing first one of the arguments that will appear in most of the functions. The argument called `detcomp` refers to the deterministic components to include in the model. Three types of regressors can be included: regular deterministic components, seasonal deterministic components, and any regressor variable previously defined by the user. This argument must be a list object with the following elements:

- `regular=c(0,0,0)`, if the first and/or second element are set equal to 1, it indicates that an intercept, and/or linear trend, respectively, are included. The third element in `regular` is a vector indicating which seasonal dummies should be included. If no seasonal dummies are desired it must be set equal to zero. For example, `regular=c(1,0,c(1,2,3))` would include an intercept, no trend, and the first three seasonal dummies.
- `seasonal=c(0,0)`, if an element is set equal to 1, it indicates that seasonal intercepts, and/or seasonal trends, respectively, are included in the model.
- `regvar=0`, if none regressor variables are considered, this object must be set equal to zero, otherwise, the names of a matrix object previously defined with the desired regressors by columns should be indicated.

We will take for the examples below the logarithms of the Real Gross Domestic Product in Germany. This time series, along with the others provided with the package, is analysed in Franses (1996). Let's start loading the package and the data.

[3] A 'pdf' version is available in the 'partsm/doc' subdirectory.

```
> library(partsm)
> data("gergnp")
> lgergnp <- log(gergnp, base=exp(1))
```

4.1. Model order selection

The first issue we will deal with is the selection of the order for the periodic autoregressive model. The function `Fnextp.test` performs a test for the significance of prospective lag parameters of order $p + 1$ in an AR(p) or PAR(p) model. It is performed as an F-statistic that sets the parameters of order $p + 1$ equal to zero. We will use this statistic and the Akaike's [AIC] and Schwarz's [BIC] information criteria to select the autoregressive order. The code below computes these statistics for PAR models of order ranged between 1 and 4.

The function `fit.ar.par` fits a PAR model with seasonal intercepts and stores the results in the `lmpar` object of class `fit.partsm`. Then, the AIC and BIC statistics are computed by means of the function `AIC` available in the **base** R-package.

Finally, the null hypothesis $\phi_{(p+1),s} = 0$ is checked with `Fnextp.test`. The F-statistics and the corresponding p-values are stored in `Fnextp` and `Fpval`, respectively. These statistics are computed for PAR models with seasonal intercepts.

```
> detcomp <- list(regular=c(0,0,0), seasonal=c(1,0), regvar=0)
> aic <- bic <- Fnextp <- Fpval <- rep(NA, 4)
> for(p in 1:4){
+   lmpar <- fit.ar.par(wts=lgergnp, detcomp=detcomp, type="PAR", p=p)
+   aic[p] <- AIC(lmpar@lm.par, k=2)
+   bic[p] <- AIC(lmpar@lm.par, k=log(length(residuals(lmpar@lm.par))))
+   Fout <- Fnextp.test(wts=lgergnp, detcomp=detcomp, p=p, type="PAR")
+   Fnextp[p] <- Fout@Fstat
+   Fpval[p] <- Fout@pval
+ }
```

To save space, the results of the code are reported in Table 1. PAR parameters of order 2 are significant, whereas, lag parameters up to order 4th appear to be equal to zero. Furthermore, the AIC and BIC criteria reach the lowest value for a second order model. Therefore, we will analyse the properties of a second order PAR model with seasonal intercepts for the `lgergnp` time series.

4.2. Test for periodic variation in the autoregressive parameters

Once a PAR model has been defined, we can check for periodicity in the autoregressive parameters of the model. Following the notation in (1), the function `Fpar.test` performs an F-test for the null hypothesis of non-periodicity, $\phi_{is} = \phi_i$ for $s = 1, 2, ..., 4$ and $i = 1, 2, ..., p$.

Table 1: Periodic autoregressive order selection

Criterion	Periodic autoregressive order			
	1	2	3	4
AIC	−661.60	−680.89	−669.84	−661.54
BIC	−636.30	−644.44	−622.31	−603.00
$F(\phi_{p+1,s} = 0)$	8.54	0.80	1.35	2.91
p-value	0.00	0.53	0.26	0.03

When the null hypothesis is imposed an AR(p) is estimated, whereas in the alternative a PAR(p) model is fitted. Then, based on the residual sum of squares of each model, the F-statistic is computed. When four seasonal intercepts are included the statistic follows an F-distribution with (3 p, n-(4+4 p)) degrees of freedom, where n is the number of observations.

```
> dcsi <- list(regular=c(0,0,0), seasonal=c(1,0), regvar=0)
> out.Fparsi  <- Fpar.test(wts=lgergnp, detcomp=dcsi, p=2)
> show(out.Fparsi)

----
  Test for periodicity in the autoregressive parameters .

  Null hypothesis: AR( 2 ) with the selected deterministic components.
  Alternative hypothesis: PAR( 2 ) with the selected deterministic components.

  F-statistic: 43.46 on 6 and 116 DF, p-value: 0 ***

  Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> dcsit <- list(regular=c(0,0,0), seasonal=c(1,1), regvar=0)
> out.Fparsit <- Fpar.test(wts=lgergnp, detcomp=dcsit, p=2)
> show(out.Fparsit)

----
  Test for periodicity in the autoregressive parameters .

  Null hypothesis: AR( 2 ) with the selected deterministic components.
  Alternative hypothesis: PAR( 2 ) with the selected deterministic components.

  F-statistic: 8.37 on 6 and 112 DF, p-value: 1.674542e-07 ***

  Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The results of the code above show that periodicity is not rejected, therefore, the test corroborates that a periodic model fits better to the data rather than an AR model, which is constrained to seasonally constant parameters. With regard to the so-called deterministic components, the reader can check that seasonal trends can be left out in the PAR(2) model. For it, type `summary(out.Fparsit)` and check the t-statistics and p-values in

the model fitted for the alternative hypothesis for the parameters denoted as `MDT..SeasTrnd.1:4`.

4.3. Diagnostic for the fitted PAR model

The function `Fsh.test` performs an F-statistic to check whether seasonal heteroskedasticity exist in the residuals of the fitted model, in this case, a PAR(2) model with seasonal intercepts.

```
> par2 <- fit.ar.par(wts=lgergnp, type="PAR", p=2, detcomp=detcomp)
> Fsh.out <- Fsh.test(res=residuals(par2@lm.par), s=frequency(lgergnp))
> show(Fsh.out)

----
  Test for seasonal heteroskedasticity .

  F-statistic: 2.77 on 3 and 121 DF, p-value: 0.04469356 *

  Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Results in the code above show that seasonal heteroskedasticity is rejected at the 5% significance level. This analysis is very limited to validate the model and the user is suggested to carry out complementary tests such as Ljung-Box test for autocorrelation, `Box.test`; Jarque-Bera test for normality, `jarque.bera.test`; or `runs.test` for randomness[4]. Nevertheless, this analysis is beyond the scope of this paper and we will move on to analyze some properties of the selected model for the `lgergnp` time series.

4.4. Eigenvalues of the estimated $\Gamma \equiv \Phi_0^{-1}\Phi_1$ matrix

Taking the multivariate representation, as in equation (2), we can get the eigenvalues of the estimated $\Gamma \equiv \Phi_0^{-1}\Phi_1$ matrix. These eigenvalues provide a first view on the prospective unit roots.

The function `PAR.MVrepr` applied on the `out.par` object of class `fit.partsm` below, shows the matrices defined is Section 2 for the multivariate representation and some complementary information.

```
> out.par <- fit.ar.par(wts=lgergnp, type="PAR", detcomp=detcomp, p=2)
> out.MV <- PAR.MVrepr(out.par)
> out.MV

----
    Multivariate representation of a PAR model.


  Phi0:

  1.000  0.000  0.000 0
```

[4]The first function is available in the **base** package provided with the standard distribution of R. The other functions can be found in the package called **tseries**.

```
-0.279  1.000  0.000 0
 0.320 -1.237  1.000 0
 0.000 -0.422 -0.637 1


 Phi1:

0 0 -0.375 1.360
0 0  0.000 0.684
0 0  0.000 0.000
0 0  0.000 0.000


 Eigen values of Gamma = Phi0^{-1} %*% Phi1:
0.965 0.036 0 0

 Time varying accumulation of shocks:

0.173 1.182 0.492 1.360
0.140 1.158 0.573 1.064
0.118 1.054 0.552 0.881
0.134 1.160 0.594 1.010
```

There is only one eigen value of the $\Gamma \equiv \Phi_0^{-1}\Phi_1$ matrix close to 1, hence, it seems that there is a single unit root . Furthermore, since this value

There are not complex eigen values in the $\Gamma \equiv \Phi_0^{-1}\Phi_1$ matrix. It seems to exist a single unit root in the model. Furthermore, the eigenvalue close to 1 suggests that it may exist the long run unit root 1. We will take again a look at this representation and will discuss the remaining of the output of the function.

4.5. Test for a single unit root

The next step is to carry out formal tests to check whether a unit root exists. As it has been mention in Section 2.2, a PAR model contains a unit root if $\prod_{s=1}^{4} \alpha_s = 1$. We have also noticed that this restriction is satisfied in two particular cases, when α_1, α_2, α_3, and α_4 are either 1 or -1. In these cases the periodic filter collapses into the filter related to the long run unit root 1, $(1 - B)$, and to the seasonal unit root -1, $(1 + B)$, respectively, and the model is known as a PAR process for a I(1) time series, PARI.

In other cases in which the restriction $\alpha_1\alpha_2\alpha_3\alpha_4 = 1$ is fulfilled, the series is said to be guided by a periodically integrated AR process, PIAR. Then, the periodically differencing filter is $(1 - \alpha_s B)$ for $s = 1, 2, ..., 4$, where α_s are inferred from the data by fitting the model (4).

Taking into account these remarks, and as Franses (1996) suggests, it is a wise analysis to test first for the non-linear restriction $\prod_{s=1}^{4} \alpha_s = 1$. If this hypothesis cannot be rejected the time series contains a unit root. To see whether the process is a PARI process with the long run seasonal unit root

1, or the seasonal unit root -1, or a PIAR process, the following hypothesis can be checked[5],

$$H0 : \alpha_s = 1 \quad \text{for } s = 1, 2, 3,$$

$$H0 : \alpha_s = -1 \quad \text{for } s = 1, 2, 3.$$

The function `LRurpar.test` performs a likelihood ratio test for the first step we desire to check. A one-side statistic is also computed as $sign(\prod_{s=1}^{4} \alpha_s - 1) * LR^{1/2}$, where $\hat{\alpha}$ are the periodic differencing filter parameters estimated under the alternative. According to this statistics, the results of the code shows that it cannot be rejected a unit root.

```
> out.LR <- LRurpar.test(wts=lgergnp, detcomp=detcomp, p=2)
> show(out.LR)

----
  Likelihood ratio test for a single unit root in a PAR model of order 2 .

  Null hypothesis: PAR( 2 ) restricted to a unit root.
  Alternative hypothesis: PAR( 2 ).

  LR-statistic: 4.9
  ---
  5 and 10 per cent asymptotic critical values:
    when seasonal intercepts are included: 9.24, 7.52.
    when seasonal intercepts and trends are included: 12.96, 10.50.

  LRtau-statistic: -2.21
  ---
  5 and 10 per cent asymptotic critical values:
    when seasonal intercepts are included: -2.86, -2.57.
    when seasonal intercepts and trends are included: -3.41, -3.12.
```

Now it is worth checking whether α_s for $s = 1, 2, 3$, can be either 1 or -1. The corresponding F-statistic for the PAR(2) model with seasonal intercepts is computed by `Fpari1.out`. The first hypothesis is considered when `type="PARI1"` is selected. According to the results below, this hypothesis, i.e. the long run unit root 1 is rejected.

```
> Fpari1.out <- Fpari.piar.test(wts=lgergnp, detcomp=detcomp, p=2, type="PARI1")
> show(Fpari1.out)

----
  Test for a parameter restriction in a PAR model .

  F-statistic: 20.83 on 3 and 114 DF, p-value: 7.884426e-11 ***

  Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

[5]Note that only three restrictions are checked, since $\alpha_s = 1$ or -1 just for $s = 1, 2, 3$ entails either $\alpha_4 = 1$ or $\alpha_4 = -1$.

The reader can check that the seasonal unit root -1 is rejected as well, as we could expect allowing for the eigen values of the estimated matrix $\Phi 0^{-1}\,\Phi_1$. For it, just select `type="PARI-1"`.

4.6. Autocorrelation function for several transformations of the original data

In this subsection we will take a look at the nonperiodic autocorrelation function. The function `acf.ext1` computes the ACF for several transformations of the data[6]. Table 2 shows the ACF of the original data, the first differenced series and the periodically differenced series, removing four seasonal intercepts in the last two cases. According to the results is sections above other transformations are not worth considering.

Table 2: Estimated autocorrelation function for several transformations of the original data

	Transformations of the original data, y_t [a]					
Lag	y_t		$\Delta\, y_t - \hat{\delta}_s$		$(1-\alpha_s\, B)\, y_t - \hat{\delta}_s$	
0	1.00	***	1.00	***	1.00	***
1	0.95	***	−0.15	.	−0.12	
2	0.90	***	−0.47	***	−0.20	*
3	0.89	***	−0.08		0.00	
4	0.89	***	0.71	***	0.37	***
5	0.84	***	−0.15	.	−0.17	.
6	0.80	***	−0.42	***	−0.10	
7	0.79	***	−0.05		0.06	
8	0.78	***	0.54	***	0.02	
9	0.74	***	−0.08		−0.07	
10	0.70	***	−0.40	***	−0.07	
11	0.69	***	−0.06		0.03	
12	0.68	***	0.49	***	0.03	
s.e. [b]	*0.09*		*0.09*		*0.09*	

[a] The following transformations are considered: y_t: Original series; $\Delta\, y_t - \hat{\delta}_s$: residuals of the first differences on four seasonal dummy variables; $(1-\hat{\alpha}_s\, B)\, y_t - \hat{\delta}_s$: residuals of the periodic differences on four seasonal dummy variables.

[b] Standard error calculated as $1/n^{1/2}$.

Significance codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1.

The ACF of the original series shows a clearly long-memory behaviour, since autocorrelations are significant for high orders. It can also be observed that the periodically differenced series performs better than the first differencing

[6] See the standard help pages to see what transformations are considered.

filter, since the latter does not remove the stochastic behaviour.

The periodically differenced series and the corresponding seasonal path are displayed by `plotpdiff`. This function applies on object of class `fit.piartsm` as `out.piar` built above. Figure 1 shows that the periodic differencing filter success in removing the stochastic behaviour, since the seasonal paths (at the bottom of the Figure) are parallel to each other and do not display trending behaviour.

4.7. Time varying impact of accumulation of shocks

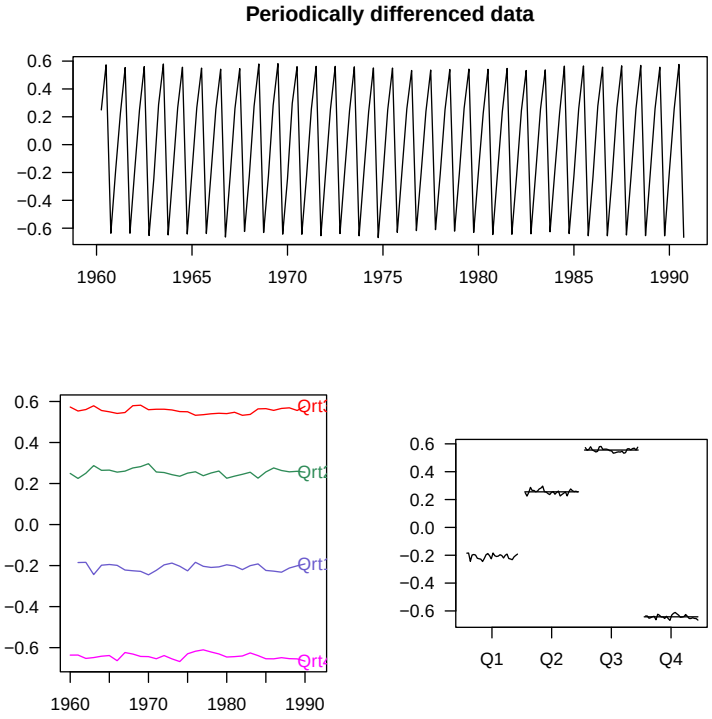
The Φ_0 and Φ_1 matrices can be used to compute the impact of accumulation of the shocks ϵ_t defined as $\Gamma\,\Phi_0^{-1}$, where Γ is $\Phi_0^{-1}\Phi_0$. That row in which the values of the impact matrix are the highest, entails that the corresponding season undergoes more severe impacts from the accumulation of all shocks. Hence, it is more likely to display fluctuations in the stochastic trend. Likewise, the column with the highest values is related to the season that has the largest long-run impact. Put in other words, the impact matrix allow the practitioner to get an idea about how the stochastic trend and the seasonal fluctuations are related.

To derive those matrices, the model (4) is estimated with `fit.piar` is used. This function requires some values to initialize the non-linear estimator. By default, initial values are computed for the non-linear model. However, in this version there may be cases in which the estimates do not converge, giving an error message. In this case, a numeric vector with initial values guessed by the user can be included. The code below let `fit.piar` to compute initial values, hence, there is no need to include a vector called `initvalues` as an argument.

```
> out.piar <- fit.piar(wts=lgergnp, detcomp=detcomp, p=2)
> out.MV <- PAR.MVrepr(out.piar)
> out.MV

----
    Multivariate representation of a PIAR model.


  Phi0:

  1.000  0.000  0.000 0
 -0.962  1.000  0.000 0
  0.000 -0.912  1.000 0
  0.000  0.000 -1.113 1


  Phi1:

 0 0 0 1.025
 0 0 0 0.000
```

```
0 0 0 0.000
0 0 0 0.000


  Eigen values of Gamma = Phi0^{-1} %*% Phi1:
1 0 0 0

  Time varying accumulation of shocks:

1.000 1.040 1.140 1.025
0.962 1.000 1.097 0.985
0.877 0.912 1.000 0.898
0.976 1.015 1.113 1.000
```

Figure 1: Periodically differenced data and seasonal paths



For the series `lgergnp`, the row with the highest values is the fourth column. This means that the stochastic trend is more likely to undergo changes in the fourth quarter. Reading the impact matrix by columns, the one with the highest is the third column, hence, the third quarter is more sensitive to changes in the stochastic trend.

Obviously, the results above also show that the PIAR model contains a root that is exactly equal to 1 because that is precisely the restriction imposed

in the parameters of the model. The other slots of the output not cited here are matrices for internal use when making forecast in a PIAR model.

4.8. Out-of-sample forecasts

The function `predictpiar` makes prediction based on PIAR model of order up to 2. By default seasonal intercepts are included. This function computes one-year-ahead forecasting on the basis of the multivariate representation. The forecast for the year $T+$ is $Y_{T+1} = \Phi_0^{-1}\,\mu + \Phi_0^{-1}\,\Phi_1\,Y_T$. The confidence intervals are computed deriving the multivariate moving averages representation from the matrices for the multivariate PIAR model.

The code below performs 24 ahead forecasts in the PIAR(2) model. For programming convenience, the number of forecasts, `hpred`, must be a multiple of the periodicity of the data. The output of this function provides an object of class `pred.piartsm` containing the forecast, the standard errors, the upper and lower 95 per cent confidence bound, as well as the input, i.e., the original data, the order of the model, and the number of predictions.

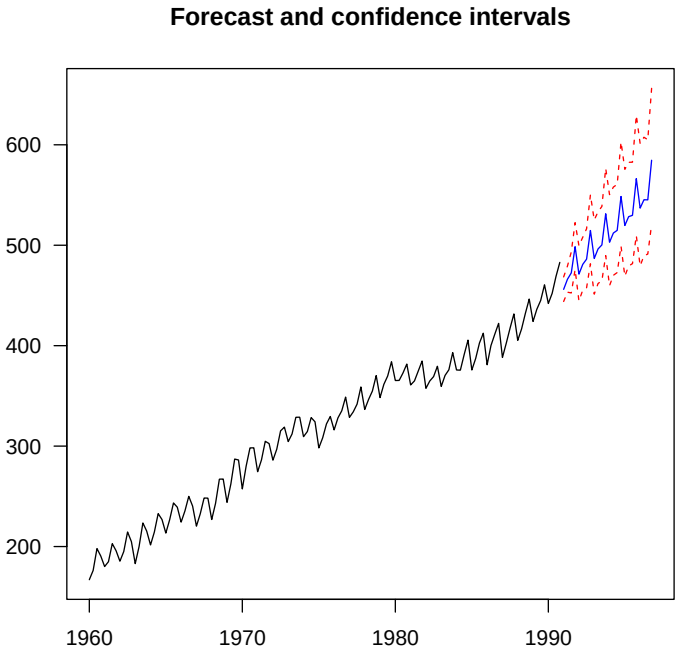
```
> out.pred <- predictpiar(wts=lgergnp, p=2, hpred=24)
> show(out.pred)

----
    Forecasts for a PIAR model of order 2 .

           fcast        fse      ucb      lcb
1991.01 6.122634 0.01367962 6.149446 6.095822
1991.02 6.144360 0.01422858 6.172248 6.116472
1991.03 6.157955 0.02203235 6.201138 6.114772
1991.04 6.211668 0.02392444 6.258560 6.164776
1992.01 6.155275 0.02971131 6.213509 6.097041
1992.02 6.175753 0.02791604 6.230469 6.121038
1992.03 6.186578 0.03080444 6.246955 6.126201
1992.04 6.243526 0.03396765 6.310103 6.176950
1993.01 6.187916 0.03867467 6.263719 6.112114
1993.02 6.207147 0.03666875 6.279017 6.135276
1993.03 6.215201 0.03766023 6.289015 6.141387
1993.04 6.275385 0.04165992 6.357038 6.193731
1994.01 6.220557 0.04589666 6.310515 6.130600
1994.02 6.238540 0.04369798 6.324188 6.152892
1994.03 6.243824 0.04344977 6.328985 6.158662
1994.04 6.307243 0.04813842 6.401594 6.212892
1995.01 6.253198 0.05212679 6.355367 6.151030
1995.02 6.269933 0.04974351 6.367431 6.172436
1995.03 6.272447 0.04855388 6.367612 6.177281
1995.04 6.339101 0.05384298 6.444634 6.233569
1996.01 6.285840 0.05768794 6.398908 6.172771
1996.02 6.301327 0.05513003 6.409382 6.193272
1996.03 6.301070 0.05317026 6.405283 6.196856
1996.04 6.370960 0.05899853 6.486597 6.255323

  'fcast': Forecast; 'fse': Forecast standard error;
```

```
'ucb': Upper confidence bound; 'lcb': Lower condidence bound.
```

Remember that the data are scaled in logarithms, hence, for the results to be interpreted, the original scale must be set. The code below makes this task and plots a graphic with the forecast and confidence intervals displayed in Figure 2.

```
> out.pred@wts <- exp(1)^out.pred@wts
> out.pred@fcast <- exp(1)^out.pred@fcast
> out.pred@ucb <- exp(1)^out.pred@ucb
> out.pred@lcb <- exp(1)^out.pred@lcb
> plotpredpiar(out.pred)
```

Figure 2: Forecast and confidence intervals



References

Chambers, J. (1998), *Programming with Data: A Guide to the S Language*, Springer-Verlag.

Franses, P. (1996), *Periodicity and Stochastic Trends in Economic Time Series*, Advanced Texts in Econometrics. Oxford University Press.

Franses, P. and Paap, R. (2004), *Periodic Time Series Models*, Advanced Texts in Econometrics. Oxford University Press.

López-de Lacalle, J. (2005), 'Periodic autoregressive time series models in `R`: The `partsm` package', BILCODEC 2005 working paper, Universidad del País Vasco UPV/EHU - Departamento de Economía Aplicada III (Econometría y Estadística).
URL: *http://econpapers.repec.org/software/ehubilcod/200501.htm*.

R Development Core Team (2011), *R: A language and environment for statistical computing*, R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.
URL: *http://www.R-project.org*.

Author(s):

Javier López-de-Lacalle.
Universidad del País Vasco.
E-mail: `javlacalle@yahoo.es`.