Software for non-stationary time series analysis and decompositions via TVAR models.

Raquel Prado

July 26, 2000

1 Overview

This software implements time-varying autoregressions or TVAR models and allows the computation of latent components in time series decompositions based on such models. In addition to model selection and computation of posterior estimates for the parameters of the TVAR model, the software supports computation of the latent processes in the decomposition of the series and the corresponding characteristic wavelengths, moduli and amplitudes related to such processes at each time.

Parameter estimation in a TVAR is carried out by specifying a Dynamic Linear Model (DLM) representation of the model West and Harrison (1997). A traditional random walk is adopted to model the evolution of the TVAR parameters over time. Specification of the observational variance and the system evolution variance-covariance matrix is handled by choosing of a couple of discount factors, as is standard in DLM literature (West and Harrison, 1997). Sequential and retrospective smoothing algorithms are then applied to compute posterior estimates of the parameters of the model.

Time series decompositions are computed following West *et al.* (1999). Additional key references in this subject are West (1997a), Prado and West (1997) and Prado (1998). At each time t the series are decomposed into a number of quasi-periodic components, each having an instantaneous characteristic frequency, modulus and amplitude, and a number of other components that usually represent noise. The frequencies, moduli and amplitudes that characterize the latent processes in the decomposition are also computed by the software.

The source code is written in Fortran 90 and it makes use of the public-domain libraries LBLAS and LAPACK (version 2.0) to perform linear algebra computations. The software is designed for use on a UNIX environment. It also provides a collection of S-Plus functions that read the output files created by the Fortran 90 programs. The S-Plus functions may then be used to graph latent components in time series decompositions based on estimated or sampled values of the TVAR model parameters. These functions also allow to graph time trajectories of the characteristic frequencies, moduli and amplitudes related to the latent components.

2 Fortran 90 programs

The Fortran 90 software consists basically of three programs: **grid90**, **tvar90** and **decomp90**. These programs make use of functions, subroutines and modules that are also written in Fortran 90. A brief description on the usage of these programs follows below.

2.1 grid90

This program computes the values of the log-likelihood and the mean squared error (MSE) for a grid of values on the model order and the two discount factors that specify the evolution of the observational variance and the evolution of the parameters of the TVAR model. The user must provide the following:

- a datafile
- number of points in the time series
- a file with the prior parameters n0,v0 and c0
- a file containing a grid of points on the model order and the discount factors or alternatively, in order to build a grid, the user should provide,
    - a maximum TVAR model order
    - a minimum TVAR model order

- the number of points in the interval (0.95,1] to define a grid on the variance discount factor
- the number of points in the interval (0.99,1] to define a grid on the system discount factor

Once a grid of values is provided, **grid90** computes the values of the joint log-likelihood function and the square root of the MSE for each combination of points and finds the optimal choices in terms of maximizing the log-likelihood and minimizing the MSE over the grid. It saves the log-likelihood and MSE values for the whole grid in the file "outputgrid".

2.2 tvar90

Computes posterior means and variances for the parameters of the TVAR model for a specific choice of the model order and the discount factors. It also allows sampling from the posterior distributions of the TVAR parameters. The user must provide the following:

- a datafile
- number of points in the time series
- a file containing the prior parameters n0,v0 and c0
- a variance discount factor in (0,1]
- a system discount factor in (0,1]
- number of time points sample from the corresponding posterior distributions of the TVAR parameters (optional)
- number of samples (optional)

The program saves the estimated posterior means for the parameters of the TVAR model at each time t in the file "postmean" and the samples from the posterior distributions at selected time points in the file "tvarsamples".

2.3 decomp90

This program computes the latent components in the decomposition and the corresponding wavelengths, moduli and amplitudes at each time t based on the model parameters estimated by **tvar90.** It orders the latent components, wavelengths, moduli and amplitudes according to a specific ordering criteria selected by the user from five possible options. The following options are available:

- amplitude: orders the components by increasing amplitude at each time t.
- wavelength: orders the components by increasing wavelength at each time t.
- moduli: orders the components by increasing moduli at each time t.
- wavelength(amp): orders the components by increasing amplitude at each time t. If the amplitude difference between two given components is less than a "tolerance" value such components will then be ordered by increasing wavelength. The user must provide either a file with the tolerance values at each time or a single tolerance value.
- wavelength(mod): orders the components by increasing moduli at each time t. If the modulus difference between two given components is less than a given "tolerance" value such components will then be ordered by increasing wavelength. The user must provide either a file with the tolerance values at each time or a single tolerance value.

The latent components, wavelengths, moduli and amplitudes ordered according to a selected criteria will be saved in the files "decomposition", "wavelengths", "moduli" and "amplitudes" respectively. The user may also choose to compute the latent components for the samples of the TVAR parameters if such samples are available. The components will be ordered by the order criteria previously chosen by the user. The 2.5%, 5.0%, 50%, 95% and 97.5% quantiles for the decomposition, wavelengths, moduli and amplitudes at selected time points will be saved in the files "quantdec", "quantmod", "quantamp" and "quantwave" respectively.

3 S-Plus functions

The S-Plus functions are contained in the file "splusfunc.s". Once this file is sourced into S-Plus the following functions should be available:

- **ampplot(n=1)**: plots the first **n** trajectories of the amplitudes associated with the first **n** latent components in decomposition.

- **decompplot(n=1)**: plots the first **n** latent components in the decomposition. The data is displayed at the bottom of the figure and then, from the bottom up, the latent components appear in the order previously selected by the user (e.g. if the components are ordered by amplitude the components with higher amplitude will appear at the bottom of the graph).

- **modplot(n=1,range.mod=c(0,1))**: plots the first **n** trajectories of the moduli associated with the first **n** latent components in the decomposition. The argument **range.mod** specifies the range for the y axis in the graph.

- **post.intervals(what='frequencies', obsperunit=1, nsampling=1, n=1, range.mod=c(0,1), range.freq=c(0,25))**: plots the first **n** characteristic frequencies, amplitudes or moduli trajectories over time and the corresponding approximate 90% and 95% posterior intervals. The function has four arguments. The first argument can be chosen to be either **'frequencies'** (default), **'amplitudes'** or **'moduli'**. The arguments **obsperunit** and **nsamling** are used to obtain the right scale in the y axis when plotting the frequency trajectories. **obsperunit** corresponds to the number of observations available per unit of time. If the full dataset was subsampled every xth observation then **nsampling=x**. In most applications we have one observation per unit of time and we use the whole data set for the analysis but this is not always the case (see EEG example in the next section). The arguments **range.mod** and **range.freq** specify the range for the y axis in the graphs.

- **save.list(obsperunit=1,nsampling=1)**: saves the moduli, amplitudes, frequencies and latent components in the decomposition in a list with arguments \$mod, \$amp, \$freq, \$decomp. As in **post.intervals**, **obsperunit** corresponds to the number of observations available per unit of time. If the full dataset was subsampled every xth observation then **nsampling=x**.

- **wavesplot(obsperunit=1,nsampling=1,n=1,range.freq=c(0,25))**: plots the first n wavelengths associated with the first n latent components in decomposition. As in **post.intervals**, **obsperunit** corresponds to the number of observations available per unit of time. If the full dataset was subsampled every xth observation then **nsampling=x**.

4 Examples

4.1 EEG data

Figure 1 (at the very bottom) displays 1,200 observations taken from an electroencephalogram (EEG) trace recorded after an ECT (electroconvulsive therapy) stimulus was administered to a patient. The original data was recorded at a rate of 256 observations per second and then subsampled every sixth observation. The data show evidence of time varying quasi-cyclical behavior: the frequency is higher at the beginning of the EEG series than towards the end so it seems reasonable to model the signal via time-varying autoregressions.

We can first choose the order of the model and the discount factors that maximize the joint likelihood function over a grid of values by running **grid90**. The file "dataeeg" contains the dataset displayed in Figure 1 and the prior parameters n0,d0 and c0 can be found in the file "priorfile.eeg". Following the notation in West and Harrison (1997), the prior parameters are chosen so that at time $t = 0$ a prior point estimate of the observational variance is given by d0/n0 while $(c_0 v_0)\mathbf{I}$, where $\mathbf{I}$ is the $p \times p$ identity matrix is a prior estimate of the system evolution variance covariance matrix. Therefore, at time $t = 0$, the parameters of a TVAR model have normal prior distribution centered at zero and with variance covariance matrix $(c_0 v_0)\mathbf{I}$. We can then run **grid90** as follows,

```
> grid90
 Input name of the data file:
dataeeg
 How many data points ?
1200
```

```
 Input name of the file with prior parameters
 (file must contain n0,d0,c0):
priorfile.eeg
 Do you have a file with a grid ? [y/n]
n
 Input maximum TVAR model order:
17
 Input minimum TVAR model order:
6
 Number of points in (0.95,1.0] for the variance
 discount factor grid:
5
 Number of points in (0.99,1.0] for the system
 discount factor grid:
10
```

The user may as well choose to have an input file with all the information requested by **grid90** and then simply type,

```
> grid90 < inputfile
```

The file "grid90.out" contains the optimal choices for the order and discount factors that maximize the joint likelihood functions and the optimal choices for the order and discount factors that minimize the MSE. In this case, for the prior parameters chosen (n0=1, d0=10, c0=10), we obtain:

```
 The order and discount factors that maximize
 the likelihood over the grid are:
Order  Obs. Disc.  Sys. Disc.  Log-likelihood  SQRT(MSE)
  12   0.9600000   0.9960000     -6318.04446  48.1789128425

 The order and discount factors that minimize
 the MSE are:
Order  Obs. Disc.  Sys. Disc.  Log-likelihood  SQRT(MSE)
  15   1.0000000   0.9950000     -6345.15591  48.0895454280
```

The file "outputgrid" contains the values of the log-likelihood function and the SQRT(MSE) for the whole grid. We can now fit a TVAR model to the EEG data using the model order and discount factors that maximize the log-likelihood. We run **tvar90** as follows:

```
>tvar90
 Input name of the datafile:
dataeeg
 How many data points ?
1200
 Input name of the file with prior parameters
 (file must contain n0,d0,c0):
priorfile.eeg
 Input TVAR model order:
12
 Input variance discount factor:
0.96
 Input system discount factor:
0.996

 Saving posterior means for the AR coefficients
 in file "postmean" ...

 Do you want to save a sample from the posterior
 distributions of the TVAR parameters
 at selected points ? [y/n]
y
 How many points ?
10
 How many samples per point ?
200
 Saving samples of the TVAR coefficients ...

 ... samples saved in file "tvarsamples"
```

The files "postmean" and "tvarsamples" will be now used by **decomp90** to compute the TVAR decomposition and the characteristic wavelengths, moduli and amplitudes:

```
> decomp90
 Computing decomposition, wavelengths,
 moduli and amplitudes ...

 Choose one of the following criteria
```

```
 to order the components:
 1. amplitude
 2. wavelength
 3. moduli
 4. wavelength(amp)
 5. wavelength(mod)
2

 Saving decomposition, wavelengths, moduli and
 amplitudes ordered by WAVELENGTH in
 the files "decomposition", "wavelengths",
 "moduli" and "amplitudes" ...

 Do you want to compute the decomposition
 for the sample [y/n]?
y

 Saving approximate posterior intervals in files
 "quantwave", "quantamp", "quantmod" and "quantdec" ...
```

We now use the S-Plus functions to produce graphs of the latent components in the decomposition and the trajectories of the characteristic frequencies, moduli and amplitudes over time. The following sequence of S-Plus commands will produce Figures 1 to 5,

```
>source('splusfunc.s')
>decompplot(n=6)
>wavesplot(obsperunit=256,nsampling=6,n=6)
>ampplot(n=6)
>modplot(n=6)
>post.intervals(obsperunit=256,nsampling=6,n=6)
```

As mentioned before, the number of observations per second in the original recording is 256 and the data was subsampled every sixth observations, therefore **obsperunit=256** and **nsampling=6**. The argument n=6 indicates the number of components for which the frequency trajectories (frequencies is the default option) will be graphed.

Figure 1 displays the data set and the first six quasi-periodic components in the decomposition ordered by increasing frequency. The components dis-

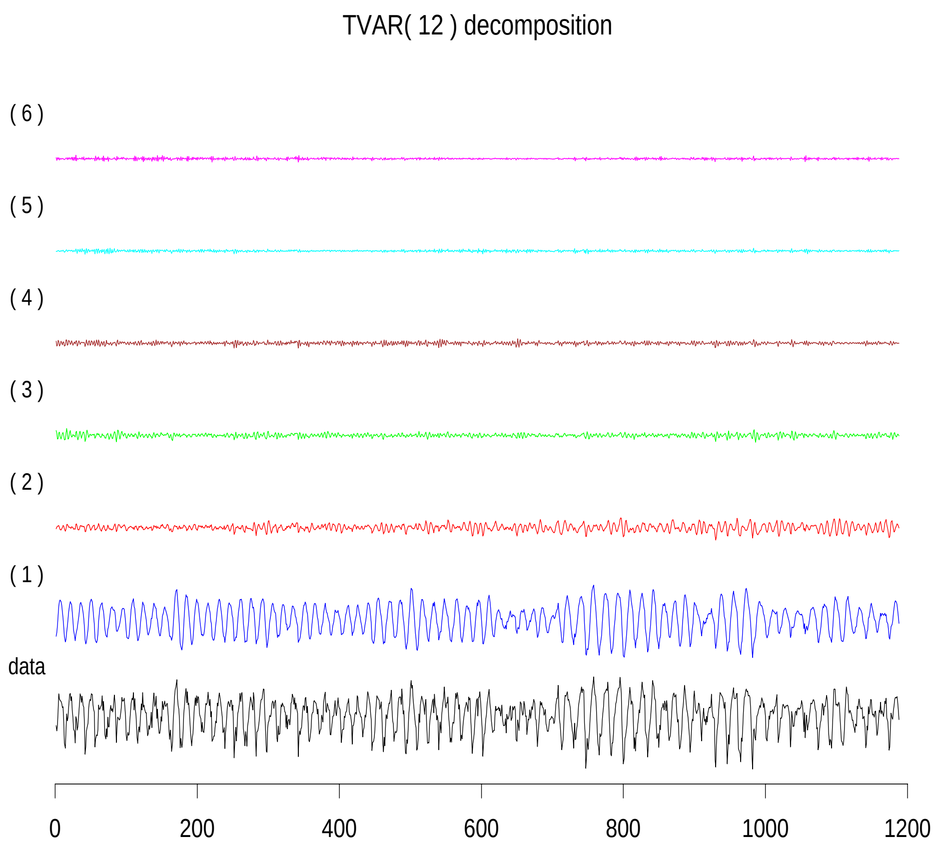

Figure 1: Latent components in the decomposition for the EEG data based on posterior mean estimates of a TVAR(12) parameter vector.

played at the bottom of the picture have lower frequencies than the components displayed at the top. In this case all the components are quasi-periodic and the trajectories of the characteristic frequencies, amplitude and moduli are shown in figures 2, 3 and 4.

Figure 5 displays the frequency trajectories over time with approximate posterior intervals at 10 time points. The points were chosen by **tvar90** and they correspond to times $t = 100$, $t = 210$, $t = 320$, $t = 430$, $t = 540$, $t = 650$, $t = 760$, $t = 870$, $t = 980$ and $t = 1090$. The approximate posterior intervals were computed with 200 samples from the posterior distributions of the TVAR parameters.

We may choose to order the components by amplitude. Figure 6 displays the trajectories of the frequencies over time ordered by amplitude. The "switching" experienced by the frequencies is due to the new ordering criteria (see discussion in West *et al.* (1999)). If we choose to order the components by wavelength(amp) with a single "tolerance" value fixed at 50 for instance, the components will then be ordered by wavelength. The amplitude differences are greater than 50 only when we compare component (1) with the rest of the components (see Figure 3), but component (1) is also the lowest frequency component.

4.2 Oxygen data

The data displayed at the bottom of figure 7 corresponds to a series of oxygen isotope measures during the last 2.5 million years. This series corresponds to a collection of series another of which has been analyzed in West (1997a) and West (1997b).

Assume we want to fit a TVAR(20) to the series . We may get the discount factors that minimize the MSE for a TVAR(20) from **grid90**:

```
> grid90
 Input name of the data file:
dataoxygen
 How many data points ?
866
 Input name of the file with prior parameters
 (file must contain n0,d0,c0):
priorfile.oxy
 Do you have a file with a grid ? [y/n]
```

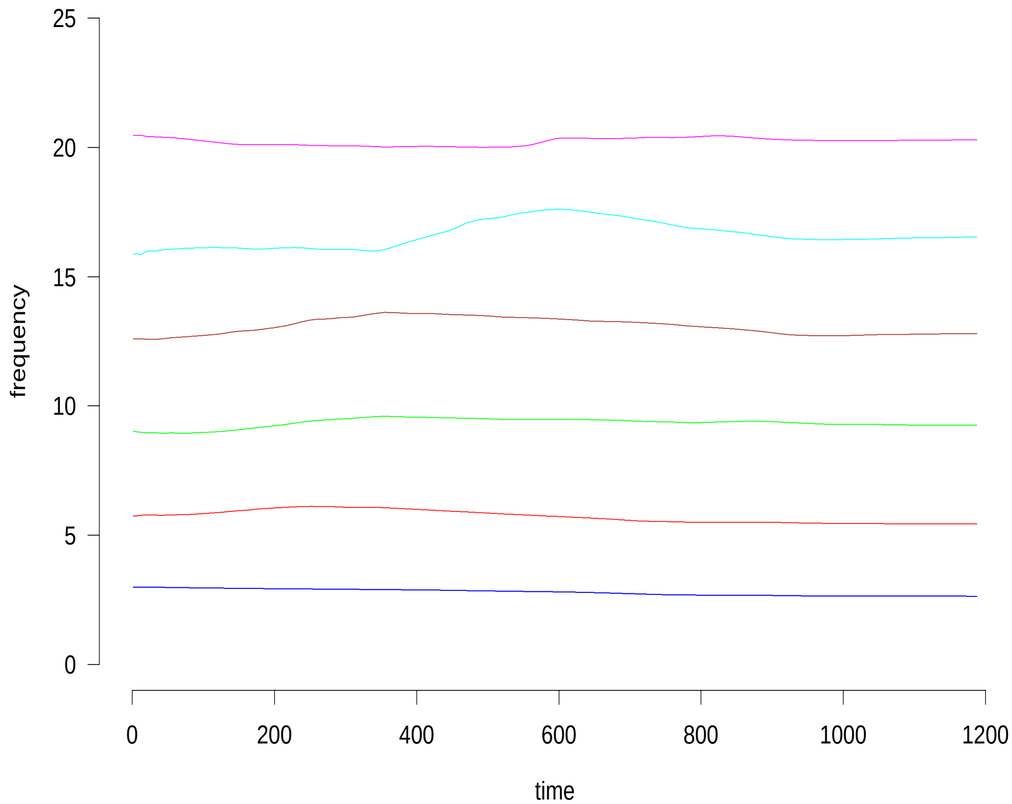

Figure 2: Estimated time trajectories of frequencies of the quasi-periodic components ordered by wavelength.

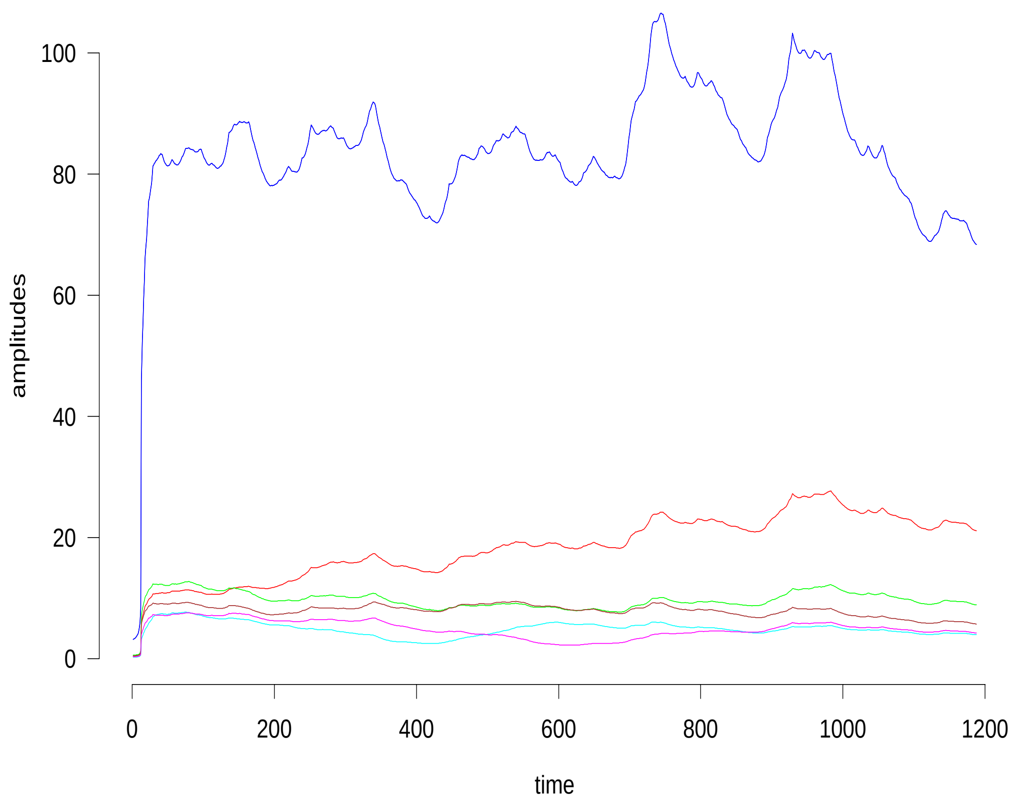

Figure 3: Estimated time trajectories of amplitudes of the quasi-periodic components ordered by wavelengths.

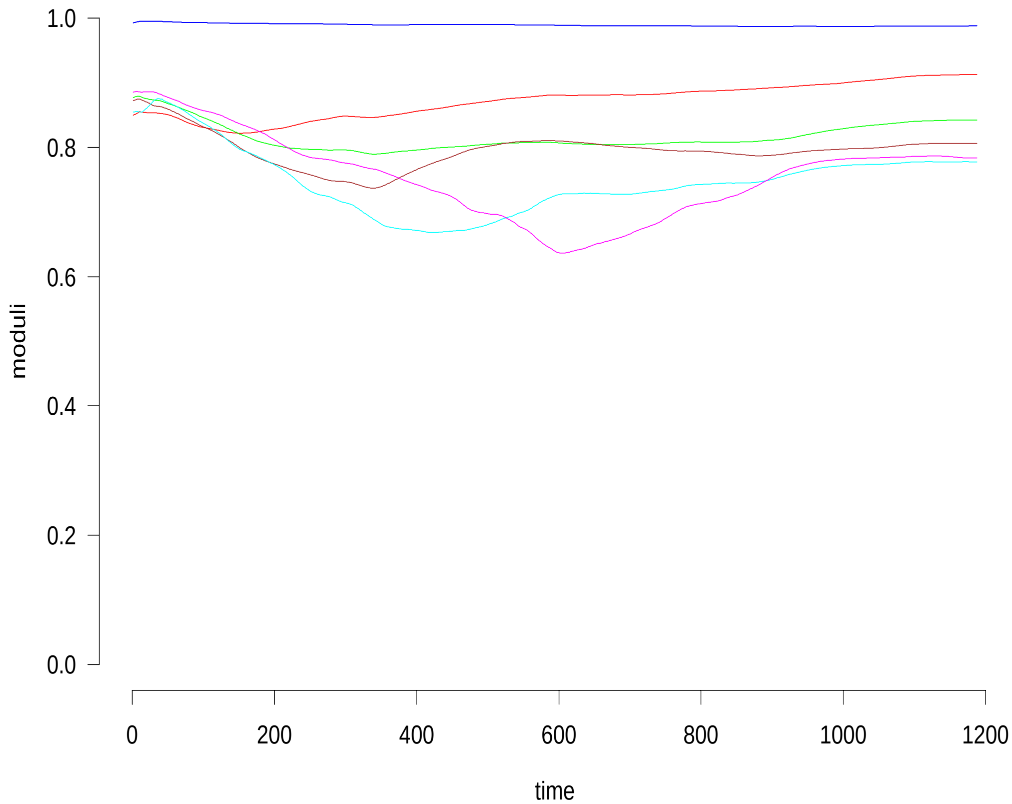

Figure 4: Estimated time trajectories of the moduli of the quasi-periodic components ordered by wavelengths.

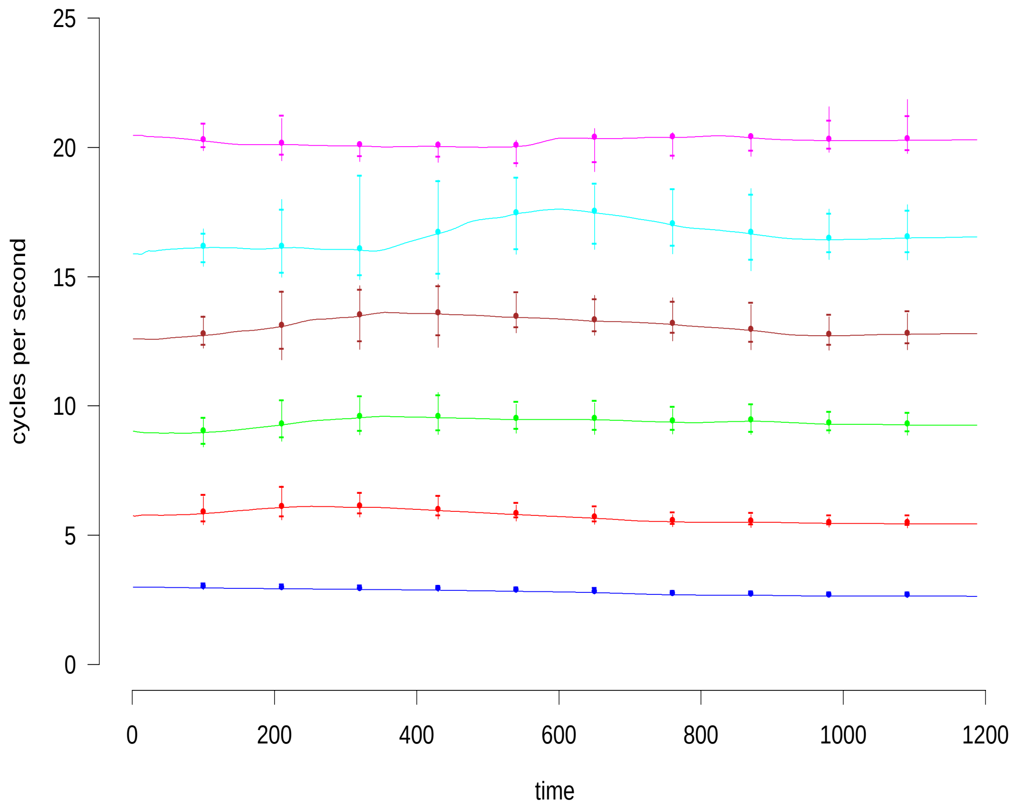

Figure 5: Estimated time trajectories of the quasi-periodic components and approximate 95% posterior intervals at selected time points

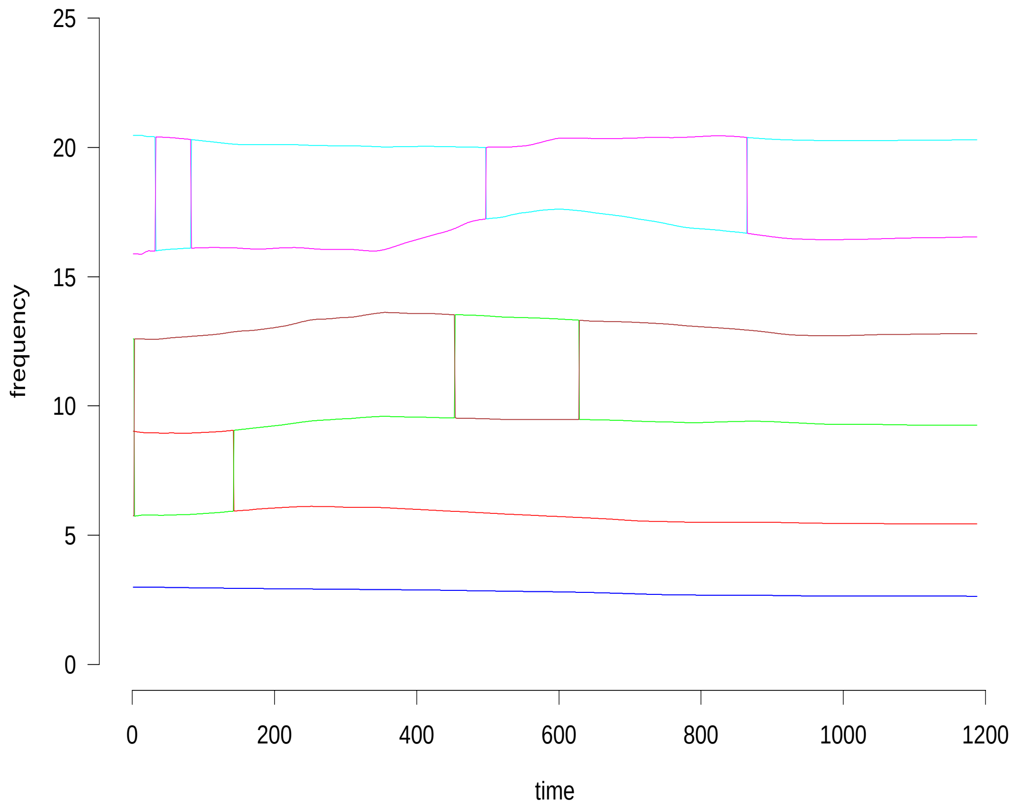

Figure 6: Estimated time trajectories of frequencies of the quasi-periodic components ordered by wavelength.

```
n
 Input the maximum TVAR model order:
20
 Input the minimum TVAR model order:
20
 Number of points in (0.95,1.0] for the variance
 discount factor grid:
10
 Number of points in (0.99,1.0] for the system
 discount factor grid:
10
```

The discount factors that minimize the MSE error are 0.97 and 0.998. Figures 7 and 8 display the first four latent components ordered by amplitude and the corresponding frequency trajectories over time. These results are based on a TVAR(20) with discount factors 0.97 and 0.998. In order to produce these pictures we simply run **tvar90**, **decomp90** and then use the S-Plus functions to graph the results,

```
> decompplot(n=4)
> wavesplot(n=4,range.freq=c(0,0.2))
```

Figure 9, display the first four latent components in the decomposition of the series but this time the latent processes are ordered by wavelength.

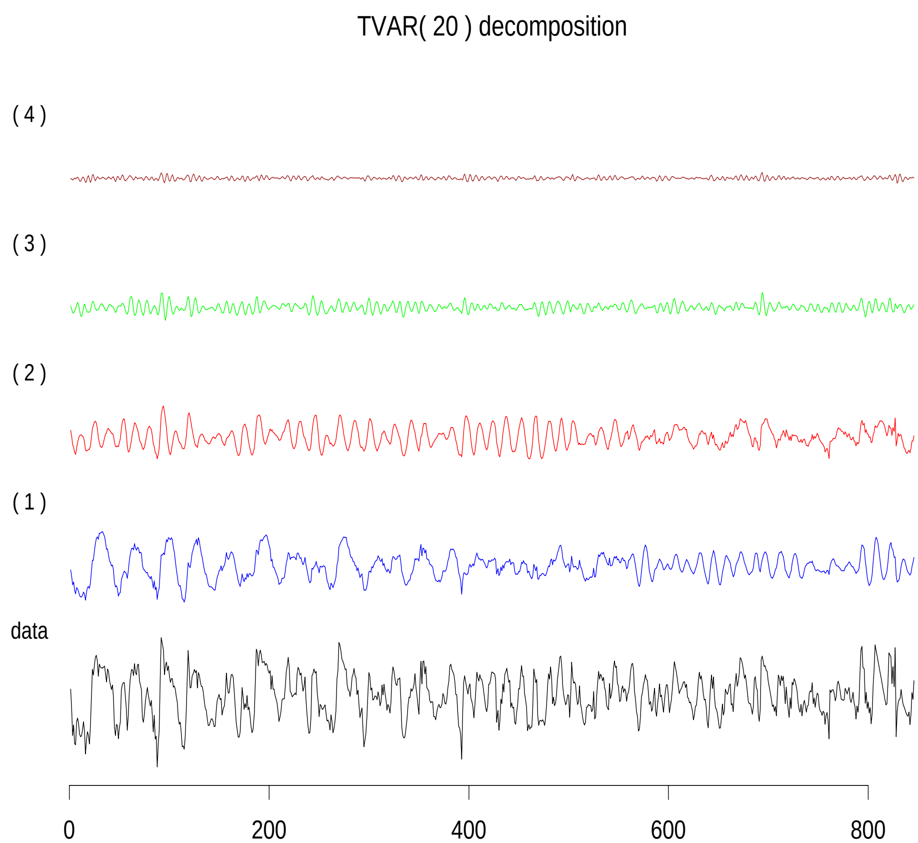

Figure 7: Estimated time trajectories of frequencies of the quasi-periodic components ordered by wavelength.

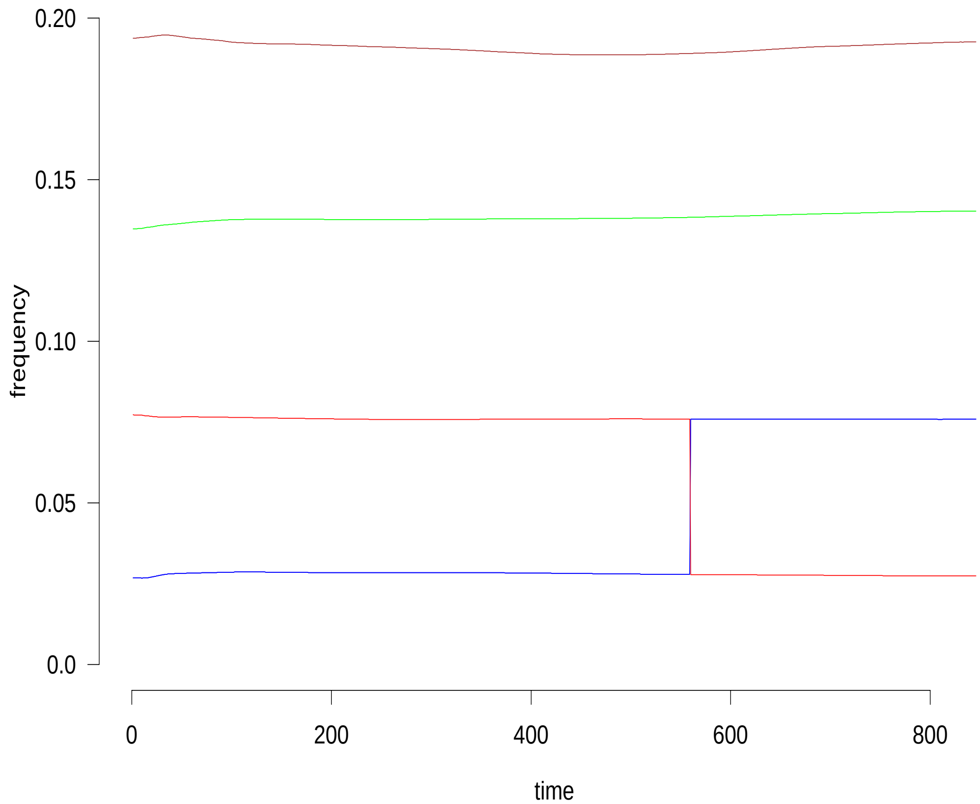

Figure 8: Estimated time trajectories of frequencies of the quasi-periodic components ordered by wavelength.

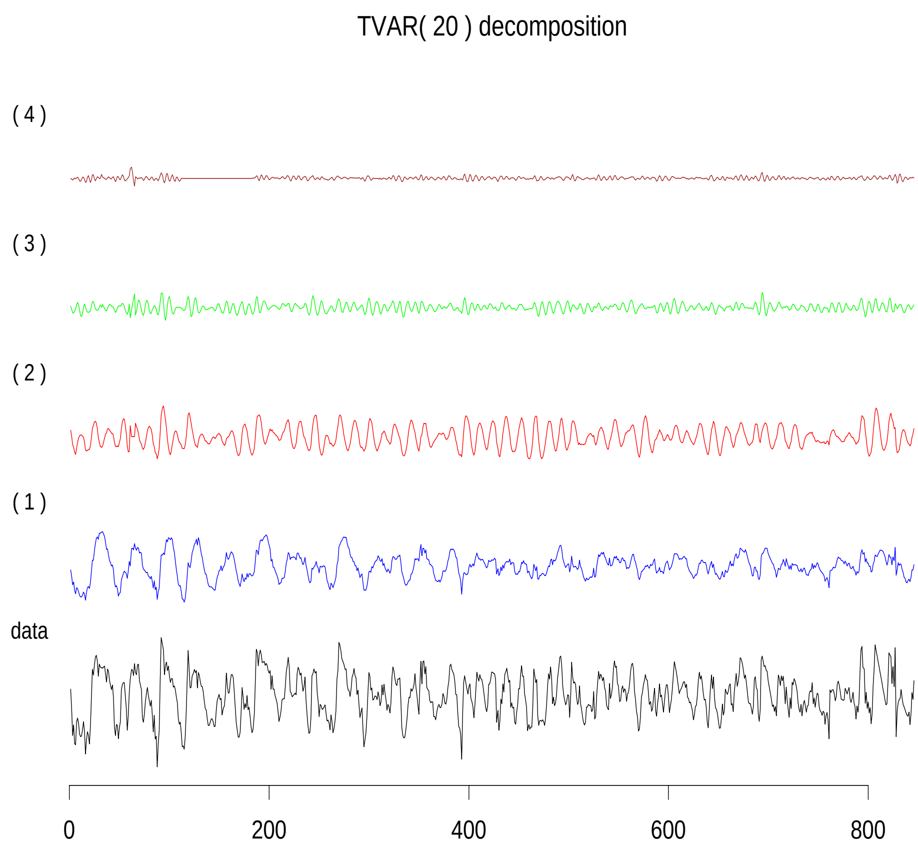

Figure 9: Estimated time trajectories of frequencies of the quasi-periodic components ordered by wavelength.

Bibliography

Prado, R. (1998) Latent structure in non-stationary time series. Ph.D. Thesis. Duke University, Durham, NC.

Prado, R. and West, M. (1997) Exploratory modelling of multiple non-stationary time series: Latent process structure & decompositions. In *Modelling Longitudinal and Spatially Correlated Data, ed: T.Gregoire*. New York: Springer-Verlag.

West, M. (1997a) Modelling and robustness issues in bayesian time series analysis (with discussion). In *Bayesian Robustness 2,* Lecture-Notes Monograph Series (eds J. Berger, F. Ruggeri and L. Wasserman), vol. 29. Hayward, CA: Institute of Mathematical Statistics.

West, M. (1997b) Time series decomposition. *Biometrika*, **84,** 489–94.

West, M. and Harrison, J. (1997) *Bayesian Forecasting and Dynamic Models*, 2nd edn. New York: Springer-Verlag.

West, M., Prado, R. and Krystal, A.D. (1999) Evaluation and comparison of EEG traces: Latent structure in nonstationary time series. *Journal of the American Statistical Association*, **94,** 1083–1095.