Time-series modeling in continuous-time
Seminar Norges Handelshøjskole

Helgi Tómasson

December 3rd, 2015

A brief presentation

- Thank you for inviting me to NHH. A pleasure to be in Bergen
- I am professor econometrics/statistics at Faculty of Economics (used to be business and economics) at the University of Iceland.
- I have a degree in applied mathematics from the University of Iceland and a Fil. Dr. from the University of Gothenburg.
- I have had a tenured position for 25 years and I have taught econometrics/statistics and financial mathematics (Ito, Black-Scholes, etc.)
- I am a computing person and I write my own programs, R, MATLAB/OCTAVE, FORTRAN, RCPP, JULIA etc. I have given a course on numerical methods in Economics and Finance (the Miranda/Fackler book).
- I have given consultations in medical statistics and real-estate appreciations modeling.
- Mostly my focus has been on time-series.

Why continuous-time modelling?

- Many phenomena in science, biology, economics, finance, etc. evolve in time.
- The variables of interest have a value at each point in time.
- Variables are frequently measured regularly, daily, monthly, quarterly, etc.
- Traditional time-series methods treat such data using discrete-time dynamics.
- The regular sampling of data and ease of computation favour the discrete-time approach.
- The parametrization of a discrete-time time-series model depends on the sampling frequency.
- The continuous-time approach is in some sense, more direct, non-synchronous data analysis is natural (no such thing as missing data). Computationally somewhat more complicated.

Plan of talk

- An illustration of an applied approach described in Tómasson (2015).
- In that paper the focus is on a linear stochastic-differential-equation.
- The ARMA approach is standard in many time-series-econometrics applications.
- Non-linear SDE are also possible.
- A few comments on R-programming and distributing a package.
- A few comments on further possibilities.
- Conclusion.

Continuous-time ARMA

- The observations are measurements of dependent random variables. The dependency structure is due to the ordering in time.
- Dynamics can be described by differential equations if time is continuous and by difference equations if time is discrete.
- In many scientific fields, physics, finance, economics, biology, etc. theoretical dynamic process are described in continuous-time by means of differential equations
- By doing the statistical modelling in continuous time we are in sense closer to the real physical model
- The probabilistic models for dynamics are the stochastic-differential-equations (SDE)

The discrete-time ARMA is of the form:

$$Y_t = \phi_1 Y_{t-1} + \cdots + \phi_p Y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \cdots - \theta_q \varepsilon_{t-q}$$
$$E(\varepsilon_t) = 0, \text{ when } t \neq s \quad E(\varepsilon_t \varepsilon_s) = 0, \; E(\varepsilon_t^2) = \sigma_\varepsilon^2.$$

- This is the ARMA(p,q) model, we call ε_t a „white-noise".
- A simple way of catching dynamics
- The famous cookbook by Box & Jenkins (1976) popularized the use of ARMA models in time series analysis.
- Here time, t, is discrete, i.e. $t = 1, 2, \ldots$
- This is a linear stochastic difference equation.

By subtracting (and adding) Y_{t-1} from the definition of the ARMA(p,q), the ARMA can be written as:

$$\Delta^p Y_t + \tilde{\phi}_1 \Delta^{p-1} Y_t + \cdots + \tilde{\phi}_p Y_t = \varepsilon_t + \cdots$$
$$\Delta = \text{ the difference operator }, \quad \Delta Y_t = Y_t - Y_{t-1}.$$

We would like to have an analogous continous-time version

Answer:

$$Y^{(p)}(t) + \alpha_1 Y^{(p-1)}(t) + \cdots + \alpha_p Y(t) =$$
$$\sigma\, \mathrm{d}(W(t) + \beta_1 W^{(1)}(t) + \cdots \beta_q W^{(q)}(t)).$$

Here $Y^{(p)}(t)$ denotes the p-the derivative of process $Y(t)$. The $\mathrm{d}W(t)$ denotes the (normal) continuous-time white-noise. $W(t)$ is the Wiener process.

Stricly mathematically speaking the white noise $\mathrm{d}W(t)$ does not exist, let alone higher derivatives. Therefore the formula defining the CARMA(p,q) is of a rather formal nature. The CARMA, however is well defined. The first p derivatives of the CARMA process $Y(t)$ exist, so the path of $Y(t)$ is "smooth".

- E.g., the CAR(1) process, (Ornstein-Uhlenbeck):

$$\mathrm{d}Y(t) = -\alpha Y(t)\,\mathrm{d}t + \sigma\,\mathrm{d}W(t), \quad \text{implies}$$
$$Y(t) = \int_{t_0}^{t} (-\alpha Y(s)\,\mathrm{d}s + \sigma\,\mathrm{d}W(s))$$

  I.e. $Y(t)$ is an integral, the pattern must be smooth.
- The AR(1) is a subset of AR(2) and all ARMA(p,q) with p$\geq$1.
- The continuous-time AR(1), CAR(1), is not a subset of CAR(2). I.e. the CAR(p) family does not form a sequence of statistically nested models.
- The family of CARMA(p,q) models is a subset of CARMA(p+1,q+1).

- To be able to generalize from a single realization of a stochastic process, some kind of stability condition is necessary. Stationarity (and ergodicity) is the formal condition in time-series analysis.
- The stationarity conditions restricts the parameter space of ARMA and CARMA models.
- The ARMA model can be stated by use of polynomial functions of the backward operator, $BY_t = Y_{t-1}$,

$$\Phi(B)\, Y_t = \Theta(B)\, \varepsilon_t,$$
$$\Phi(z) = 1 - \phi_1\, z - \cdots - \phi_p\, z^p,$$
$$\Theta(z) = 1 - \theta_1\, z - \cdots - \theta_q\, z^q.$$

- Similarly the CARMA model can be stated by use of polynomial functions of the differential operator D, $D\,Y(t) = Y'(t)$.

$$
\begin{gathered}
Y^{(p)}(t) + \alpha_1 Y^{(p-1)}(t) + \cdots + \alpha_p Y(t) = \\
\sigma\,\mathrm{d}(W(t) + \beta_1 W^{(1)}(t) + \cdots \beta_q W^{(q)}(t)), \\
a(D)Y(t) = \sigma\,\mathrm{d}\,b(D)W(t), \\
a(z) = z^p + \alpha_1\, z^{p-1} + \cdots + \alpha_p, \\
b(z) = 1 + \beta_1\, z + \cdots + \beta_q\, z^q.
\end{gathered}
$$

- The ARMA process is stationary if the roots of $\Phi(z)$ are outside the unit circle.
- The CARMA process is stationary if the roots of $a(z)$ have negative real parts.
- Enforcing the stationarity restriction is complicated if $p > 2$.
- Routh-Hurwitz algorithm is a well known (100+ years old) algorithm for checking the condition of $a(z)$.
- In this work a method a method based on Pham & Breton (1991) is used.
- Another possibility is to combine the conditions for $\Phi(z)$ using methods described in Monahan (1984) and methods described in Belcher, Hampton & Tunnicliffe Wilson (1994).

Estimation and simulation methods

- Discrete obersvations of a continuous process are of the form:

$$y(t_1), y(t_2), \ldots$$

- The likelihood function of a CARMA is calculated by use of state space representation and the Kalman filter.
- The log-likelihood function can be maximized of the space of stationary CARMA by using a numerical optimization routine.
- If a candidate, $\hat{f}(\omega)$, for the spectral density is available another way would be to use a Whittle-type estimator and minimize something like:

$$\min_{\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma} \int_{-\infty}^{\infty} (\log(f(\omega)) + \hat{f}(\omega)/f(\omega))\, \mathrm{d}\omega,$$

replacing the integral with a sum over "appropriate" frequencies.

- Simulation of CARMA in the time domain is straightforward by using state space and Kalman filter.
- A frequency domain representation of CARMA:

$$
\begin{aligned}
Y(t) &= \int_{-\infty}^{\infty} \exp(i\omega\, t)\, \mathrm{d}Z(\omega), \\
E(\mathrm{d}Z(\omega)\overline{\mathrm{d}Z(\omega)}) &= f(\omega)\, \mathrm{d}\omega, \\
E(\mathrm{d}Z(\omega)) &= 0, \\
E(\mathrm{d}Z(\omega)\,\overline{\mathrm{d}Z(\lambda)}) &= 0, \quad \lambda \neq \omega.
\end{aligned}
$$

can be use to motivate frequency domain simulation approaches.

Then, an interval $(-\omega_c, \omega_c)$, that represents a high proportion of the variability in $Y(t)$ is chosen. The interval $(0, \omega_c)$ is then divided into M subintervals with $\Delta_i = (\omega_i - \omega_{i-1})$. A classical approach is that of Rice (1954):

$$Y_{Rice}(t) = \sum_{i=1}^{M} 2\sqrt{f(\omega_i)\Delta_i} \cos(\omega_i t - U_i),$$
$$\text{with } U_i \text{ independent } U(-\pi, \pi).$$

Sun & Chaika (1997) give a modified version:

$$Y_{SC}(t) = \sum_{i=1}^{M} R_i \cos(\omega_i t - U_i), \text{ with } U_i \text{ independent } U(-\pi, \pi),$$
$$\text{and}$$
$$R_i \text{ independent Rayleigh with } E(R_i^2) = 4f(\omega_i)\Delta_i.$$

The simulated processes $Y_{Rice}(t)$ and $Y_{SC}(t)$ have the same second order properties as a theoretical normal $Y(t)$ with spectral density $f(\omega)$. $Y_{SC}(t)$ is normally distributed, whereas $Y_{Rice}(t)$ is only approximately normal.

Some numerical issues

- Inverting matrices: Solved by slightly modified LAPACK FORTRAN routines.
- Matrix exponential: Solved by FORTRAN subroutines by Sidje (1998).
- Fourier transform for non-uniform time scale: Solved by FORTRAN subroutines by Greengard & Lee (2004).
- Various numerical tasks, maximizations, random-numer generations, etc: Solved by various R-packages, MASS, PolynomF, msm, nlme, numDeriv, (R Development Core Team, 2011).

On state space representations

Imagine a second order linear differential equation:

$$x''(t) + a_1 x'(t) + a_2 x(t) = 0, \quad \text{can be written as}$$

$$\mathbf{z}(t) = \begin{bmatrix} x(t) \\ x'(t) \end{bmatrix}$$

$$\mathbf{z}'(t) + A\mathbf{z}(t) = 0, \quad \text{with}$$

$$A = \begin{bmatrix} 0 & -1 \\ a_2 & a_1 \end{bmatrix}$$

$$\mathbf{z}'(t) + A\mathbf{z}(t) = \begin{bmatrix} x'(t) \\ x''(t) \end{bmatrix} + \begin{bmatrix} -x'(t) \\ a_2 x(t) + a_1 x'(t) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

A state space representation of ARMA(p,q)

Let $\boldsymbol{\beta} = (1, \theta_1, \dots, \theta_q)$ and

$$\boldsymbol{X}_t = \begin{bmatrix} X_{t-p+1} \\ X_{t-p+2} \\ \vdots \\ X_t \end{bmatrix}, \quad T_t = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ \phi_p & \phi_{p-1} & \phi_{p-2} & \cdots & \phi_1 \end{bmatrix},$$

$$\boldsymbol{R} = [0 \quad 0 \quad \cdots \quad 0 \quad 1]'$$

$$\boldsymbol{X}_{t+1} = T\boldsymbol{X}_t + \boldsymbol{R}\varepsilon_{t+1}, Y_t = \boldsymbol{\beta}'\boldsymbol{X}_t$$

$$\Delta\boldsymbol{X}_{t+1} = \boldsymbol{X}_{t+1} - \boldsymbol{X}_t = (T - I)\boldsymbol{X}_t + \text{noise} = \boldsymbol{A}\boldsymbol{X}_t + \text{noise}$$

see, e.g., (Brockwell & Davis, 1991) chatper 12. Y_t is the measured process.

A state space representation of CARMA(p,q)

$$Y(t)^{(p)} + \alpha_1 Y(t)^{(p-1)} + \cdots + \alpha_p Y(t) =$$
$$\sigma(W(t)^{(1)} + \beta_1 W(t)^{(2)} + \cdots + \beta_q W(t)^{(q)})$$

What does this mean? In state-space form

$$Y(t) = \boldsymbol{\beta}' \boldsymbol{X}(t)$$
$$\boldsymbol{\beta}' = (1 \quad \beta_1 \cdots \beta_q \quad \mathbf{0})$$
$$d\boldsymbol{X}(\boldsymbol{t}) = \boldsymbol{A}\boldsymbol{X}(t) + \sigma \boldsymbol{R} dW(t)$$

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -\alpha_p & -\alpha_{p-1} & \cdots & \cdots & -\alpha_1 \end{bmatrix}, \quad \boldsymbol{X}(t) = \begin{bmatrix} X(t) \\ X(t)^{(1)} \\ \vdots \\ X(t)^{(p-2)} \\ X(t)^{(p-1)} \end{bmatrix}$$

A state-space representation of the CARMA(p,q) process $Y(t)$.

The dynamics of the state vector:

$$d\boldsymbol{X}(\boldsymbol{t}) = \boldsymbol{A}\boldsymbol{X}(t) + \sigma \boldsymbol{R} dW(t)$$

is a linear stochastic differential equation which is easy to solve.

$$\boldsymbol{X}(t) = \underbrace{\exp_M^{\boldsymbol{A}(t-t_0)} \boldsymbol{X}(t_0)}_{\text{prediction}} + \underbrace{\sigma \int_{t_0}^{t} \exp_M^{A(t-u)} \boldsymbol{R} dW(u)}_{\text{innovation}}$$

- The process is stationary if the eigenvalues of A have a negative real-part.
- The variance matrix of the innovation, Σ_{t-t_0} solves the equations:

$$A\Sigma_{t-t_0} + \Sigma_{t-t_0} A' = \sigma^2 (\exp_M^{A(t-t_0)} \boldsymbol{R}\boldsymbol{R}' \exp_M^{A'(t-t_0)} - \boldsymbol{R}\boldsymbol{R}')$$

- Shoji & Ozaki (1998) derived this form of the equation and Tsai & Chan (2000) give an algorithm for calculating Σ_{t-t_0}.

A review of spectral theory

ARMA process can be approximated by a weighted sum of trigonometric functions, with random weights.

$$Y_t \simeq \sum_{j=1}^{n} A(\lambda_j) e^{it\lambda_j}$$

$$-\pi < \lambda_1 < \cdots < \lambda_n < \pi$$

$$A(\lambda_j) \text{ uncorrelated complex random variables} E(A(\lambda_j)) = 0,$$

$$E(A(\lambda_j)A(\bar{\lambda}_j)) = \sigma_j^2$$

$$\sigma_j^2 \text{ variation due to the frequency } \lambda_j,$$

$$F(\lambda) = \sum_{\lambda_j < \lambda} \sigma_j^2 \text{ is called the spectral CDF for } Y_t,$$

On spectral properties of CARMA

- A stationary process can be represented by a stochastic integral:

$$Y(t) = \int_{-\infty}^{\infty} e^{i\omega t} f(\omega)^{1/2} \, \mathrm{d}Z(\omega)$$

where $\mathrm{d}Z(\omega)$ are uncorrelated complex random variables (e.g. a complex white-noise process, i.e. in the case of normality, Z is a standard complex Wiener-process.), such that $E(\mathrm{d}Z(\omega) = 0$, $E(\mathrm{d}Z(\omega)\mathrm{d}\bar{Z}(\omega)) = 1$, and $E(\mathrm{d}Z(\omega)\mathrm{d}\bar{Z}(\lambda)) = 0$, if $\omega \neq \lambda$. The variance of the process is given by:<

$$V(Y(t)) = \int_{-\infty}^{\infty} f(\omega) \, \mathrm{d}\omega = 2 \int_{0}^{\infty} f(\omega) \, \mathrm{d}\omega.$$

The CARMA process is frequently expressed with help of polynomials and the differential operator D, $DY(t) = Y'(t)$.

$$a(z) = z^p + \alpha_1 z^{p-1} + \cdots + \alpha_p z,$$
$$b(z) = 1 + \beta_1 z + \cdots + \beta_q z^q,$$
$$a(D)Y(t) = \sigma b(D) dW(t).$$

Then the spectral density $Y(t)$ will be:

$$f(\omega) = \frac{\sigma^2}{2\pi} \frac{b(i\omega)b(-i\omega)}{a(i\omega)a(-i\omega)}$$

A characterization of the CARMA(p,q) process is that the spectum is a rational function. The numerator in $f(\omega)$ has to be two degrees lower than the denominator, i.e., $q \leq p - 1$.

The *ctarma package I am still a beginner*

- A collection of my own R and FORTRAN programs with help from EXPOKIT, LAPACK and NUFFT.
- A highly experimental version is out. Documenation very lousy.
- The package aims to simulate and estimate CARMA(p,q) models.
- It is a beginner's project. Hints and suggestions welcome. How to write documentation, how to organize history/changelog files etc. Also academic suggestions and practical applications, etc.
- Due to licensing issues I have now removed dependency on EXPOKIT an NUFFT.

Illustration

```
#
#  Load ctarma package
#
  > library(ctarma)
Loading required package: MASS
Loading required package: PolynomF
Loading required package: msm
Loading required package: nlme
Loading required package: numDeriv
ctarma loaded
#
# Define a CARMA(2,1) model
#
> a=c(2,40)
> b=c(1,0.15)
> sigma=6.5
```

```
#
# Simulate, exponential sampling times
#
> tt=cumsum(rexp(1000))/10

> y=carma.sim.timedomain(tt,a,b,sigma)
#
# Define a reference model, CAR(1), Ornstein-Uhlenbeck
#
> m0=ctarma(ctarmalist(y,tt,1,1,1))
#
# Estimate the CAR(1)
#
m0e=ctarma.maxlik(m0)
```

```
#
# Expand CARMA(1,0) to an equivalent CARMA(2,1)
#
> m21=ctarma.new(m0e)
#
> ctarma.loglik(m21)
[1] -326.4575
> ctarma.loglik(m0e)
[1] -326.4575
> m21$ahat
[1] 3.863464 2.863464
> m21$bhat
[1] 1 1
> m21$sigma
[1] 1.619050
```

```
#
# Estimate the CARMA(2,1)
#
> m21e=ctarma.maxlik(m21)

> summary(m21e)
$coeff
                MLE     STD-MLE
AHAT_1    2.1342112 0.25486121      (true value 2)
AHAT_2   41.1274657 1.72153107      (true value 40)
B_0       1.0000000 0.00000000
BHAT_1    0.1379970 0.02623775      (true value 0.15)
SIGMAHAT  6.9956494 0.65072796      (true value 6.5)

$loglik
[1] -52.97566                       (log-likelihood for
                                     CAR(1) -326.475)

$bic
[1] 87.51444

>
```

```
#
#  Frequency domain estimation is also possible
#
> m21w=ctarma.whittle(m21,w=(0:2000)/100)
#
#
> ctarma.loglik(m21w)
[1] -77.18719
$coeff
                MLE    STD-MLE
AHAT_1    3.3292393 0.26239118
AHAT_2   40.8512040 0.06122925
B_0       1.0000000 0.00000000
BHAT_1    0.1823049         NA
SIGMAHAT  6.0560132         NA

$loglik
[1] -77.18719
```

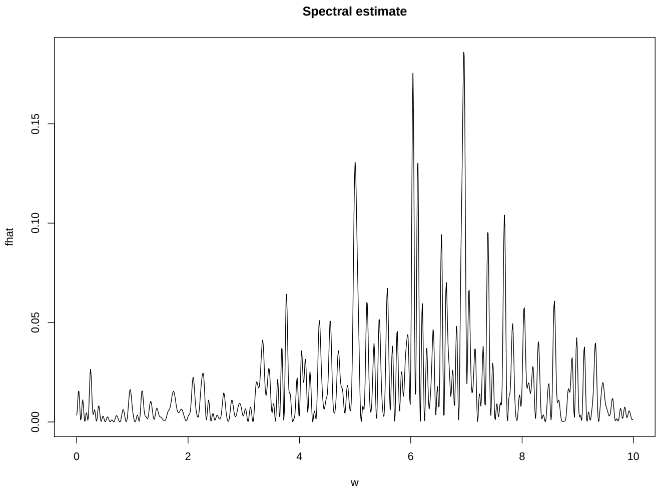


Figure: Empirical spectrum based on NUFFT

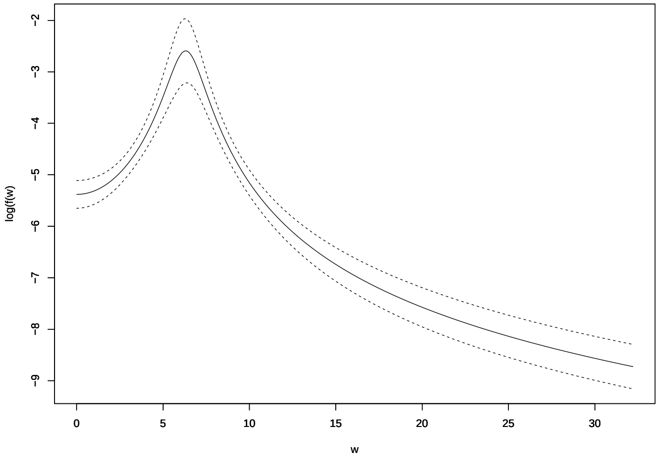


Figure: The log-spectrum based on CARMA(2,1) estimates and 95% confidence interval.

	α_1	α_2	β_1	σ
Original time-scale	2	40	0.15	6.5
Time multiplied by 10	0.2	0.4	1.5	0.206
Time multiplied by 0.1	20	4000	0.015	205.55

Table: Impact of scaling of time on CARMA parameters.

```
> ctarma.scaletime(m21,10)$ahat
[1] 0.2 0.4
> ctarma.scaletime(m21,0.1)$ahat
[1]   20 4000
#
#   Multiplying time by a constant transforms the
#   parameter values
#
> ctarma.loglik(ctarma.scaletime(m21true,10))
[1] -53.31498
> ctarma.loglik(ctarma.scaletime(m21true,0.1))
[1] -53.31498
```

Randomly spaced observations of a cycle

$$f(\omega) = \frac{{\sigma_*}^2}{4\pi}\left(\frac{1}{(\omega_c+\omega)^2+a^2}+\frac{1}{(\omega_c-\omega)^2+a^2}\right)$$

This 3 parameter models is a 4 parmeter CARMA(2,1)

$$Y^{(2)}(t) + 2aY^{(1)}(t) + (a^2 + \omega_c^2)Y(t) =$$

$$\sigma_* \sqrt{((\omega_c^2 + a^2)}\,\mathrm{d}\left(W^{(0)}(t) + \frac{W^{(1)}(t)}{\sqrt{\omega_c^2 + a^2}}\right)$$

$$\alpha_1 = 2a, \quad \alpha_2 = a^2 + \omega_c^2, \quad \beta_1 = 1/\sqrt{\omega_c^2 + a^2},$$

$$\sigma = \sigma_* \sqrt{((\omega_c^2 + a^2)} = \sigma^*/\beta_1.$$

	a_1	a_2	b_1	σ
estimate - $\bar{\Delta} = 1$	1.963	40.439	0.146	6.521
estimate - $\bar{\Delta} = 4$	1.851	39.178	0.145	6.348
s.e. - $\bar{\Delta} = 1$	0.085	0.673	0.013	0.359
s.e. - $\bar{\Delta} = 4$	0.121	1.032	0.031	0.726

Table: Exponential sampling of 5000 observations of a cyclical CARMA(2,1) (one replication).

$$a = \sigma * = 1, \quad \omega_c = 2\pi(\text{ radians per time unit }) =$$
$$\text{1 cycle per time unit}$$
$$\alpha_1 = 2, \quad \alpha_2 = 1 + (2\pi)^2 = 40.478, \quad \beta_1 = 0.157, \sigma = 6.362.$$

A real data example: Average temperature on Earth

Jouzel & et al. (2007) show data for average temperature on Earth for the past 800.000 years. An unevenly sampled time series with about 5000 observations.

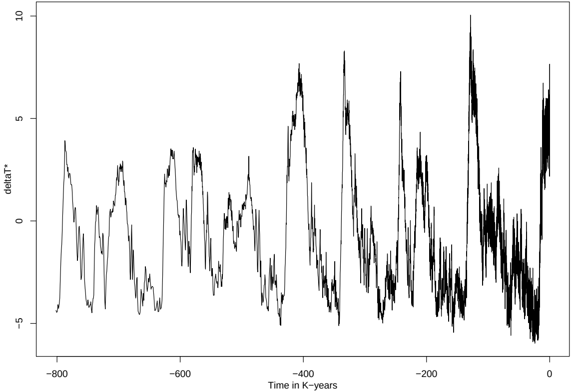


Figure: History of the Earth's temperature.

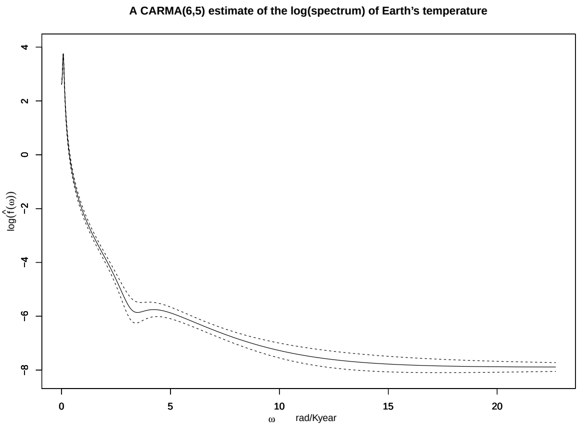
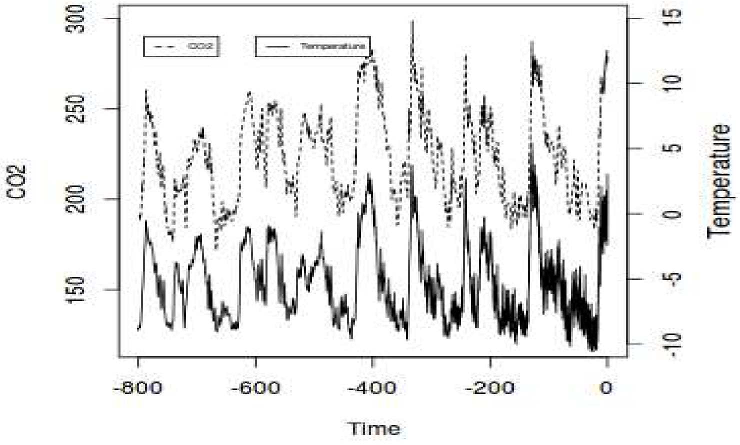
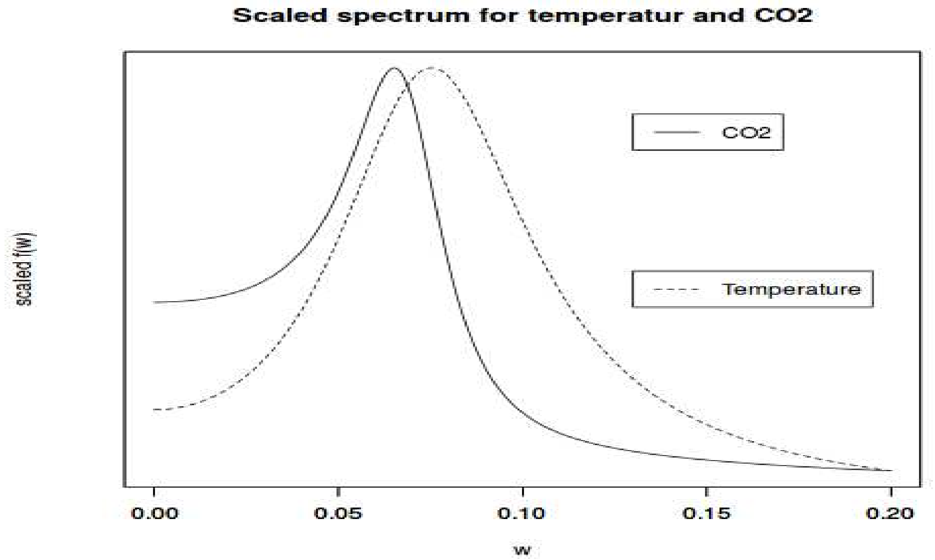


Figure: The log-spectrum of an estimated CARMA(6,5).

What about CO_2? Are these series dependent?



Are these series related (trend was removed)?



Dicussion

- Non-linear generalizations are possible (if very good knowledge of the nature of non-linearity is available)
- Non-Gaussian innovations might be an alternative to non-linear models.
- The package was written in FORTRAN (for speed) with an R-front for usability.
- It was trown out of CRAN because of license issues and that the examples did not run on Ripley's solaris machine.
- A newer version based on Rcpp is available.
- I might submit it to CRAN, R-forge or start a git-hub for distributing my packages. I have JULIA versions, and extended precision versions on the way.

Further possibilities

- A multivariate version is on the way.
- A Bayesian implementation based on smoothness-priors as well.
- Application to tick-data, real-estate valuation, competition in fuel markets are progressing (slowly) on my desk.
- More ideas welcome.

Conclusion

- The continuous-time approach is feasible in science, macro-economics and finance.
- I think it is also feasible in micro-data, e.g. on firms. It is conceivable that firms share a dynamic structure, but we only observe few cycles or even fractions of cycles per firm.
- THANK YOU

Belcher, J., Hampton, J., & Tunnicliffe Wilson, G. (1994). Parameterization of continuous autoregressive models for irregularly sampled time series data. *Journal of the Royal Statistical Association, series B*, *56*(1), 141–155.

Box, G. E. P. & Jenkins, G. M. (1976). *Time Series Analysis: Forecasting and Control*. Holden Day, San Fransisco.

Brockwell, P. J. & Davis, R. A. (1991). *Time Series: Theory and Methods*. Springer-Verlag.

Greengard, L. & Lee, J.-Y. (2004). Accelerating the nonuniform fast fourier transform. *SIAM Review*, *46*(3), 443–454.

Jouzel, J. & et al. (2007). Epica dome c ice core 800kyr deuterium data and temperature estimates. IGBP PAGES/World Data Center for Paleoclimatology Data Contribution Series # 2007-091. NOAA/NCDC Paleoclimatology Program.

Monahan, J. F. (1984). A note on enforcing stationarity in autoregressive-moving average models. *Biometrika*, *71*(2), pp. 403–404.

Pham, D. T. & Breton, A. L. (1991). Levinson-Durbin-type algorithms for continuous-time autoregressive models and applications. *Mathematics of Control, Signals, and Systems*, *4*(1), 69–79.

R Development Core Team (2011). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. ISBN 3-900051-07-0.

Rice, S. (1954). *Mathematical analysis of random noise. Monograph B-1589*. Bell Telephone Labs Inc., New York.

Shoji, I. & Ozaki, T. (1998). A statistical method of estimation and simulation for systems of stochastic differential equations. *Biometrika*, *85*, 240–243.

Sidje, R. B. (1998). Expokit. A software package for computing matrix exponentials. *ACM Trans. Math. Softw.*, *24*(1), 130–156.

Sun, T. & Chaika, M. (1997). On simulation of a Gaussian stationary process. *Journal of Time Series Analysis*, *18*(1), 79–93.

Tómasson, H. (2015). Some computational aspects of gaussian carma modelling. *Statistics and Computing*, *25*(2), 375–387.

Tsai, H. & Chan, K. (2000). A note on the covariance structure of a continuous-time ARMA process. *Statistica Sinica*, *10*, 989–998.