Bayesian analysis of non-Gaussian Long-Range Dependent processes

Tim Graves[1], Christian Franzke[2], Nick Watkins[2], and Bobby Gramacy[3]

Statistical Laboratory, University of Cambridge, Cambridge, UK

British Antarctic Survey, Cambridge, UK

Booth School of Business, The University of Chicago, Chicago, USA

May 18, 2012

Outline

Outline

Long Range Dependence

Definition

A Long Range Dependent process is a stationary process for which

$$\sum_{k=-\infty}^{\infty} \rho(k) = \infty.$$

Long Range Dependence

Definition

A Long Range Dependent process is a stationary process for which

$$\sum_{k=-\infty}^{\infty} \rho(k) = \infty.$$

... [T]he stationary long memory processes form a layer among the stationary processes that is "near the boundary" with non-stationary processes, or, alternatively, as the layer separating the non-stationary processes from the "usual" stationary processes. [Samorodnitsky, 2006]

Long Range Dependence

Definition

A Long Range Dependent process is a stationary process for which

$$\rho(k) \sim ck^{2d-1}, \quad 0 < d < \frac{1}{2}.$$

Long Range Dependence

Definition

A Long Range Dependent process is a stationary process for which

$$\rho(k) \sim ck^{2d-1}, \quad 0 < d < \frac{1}{2}.$$

Definition

A Long Range Dependent process is a stationary process for which

$$X_t = \sum_{k=0}^{\infty} \psi_k \epsilon_{t-k},$$

$$\psi_k \sim ck^{d-1}, \quad 0 < d < \frac{1}{2}.$$

ARFIMA processes

Definition

A process $\{X_t\}$ is an ARFIMA(p, d, q) process if it is the solution to:

$$\Phi(B)(1-B)^d X_t = \Theta(B)\varepsilon_t,$$

$$\text{where} \quad \Phi(z) = 1 + \sum_{j=1}^{p} \phi_j z^j \quad \text{and} \quad \Theta(z) = 1 + \sum_{j=1}^{q} \theta_j z^j,$$

and the innovations $\{\varepsilon_t\}$ are iid with 0 mean and variance $\sigma^2 < \infty$.
We say that $\{X_t\}$ is an ARFIMA(p, d, q) process with mean μ, if $\{X_t - \mu\}$ is an ARFIMA(p, d, q) process.

ARFIMA parameters

- μ – location parameter
- σ – scale parameter
- d – long memory parameter (long memory process iff $0 < d < 0.5$)
- ϕ – p-dimensional short memory parameter
- θ – q-dimensional short memory parameter

Which parameters are of interest?

When considering *long memory* processes, we are usually primarily interested in the parameter d (and possibly μ). The parameters σ, ϕ, θ (and even p, q) are essentially nuisance parameters.

ARFIMA parameters

- μ – location parameter
- σ – scale parameter
- d – long memory parameter (long memory process iff $0 < d < 0.5$)
- ϕ – p-dimensional short memory parameter
- θ – q-dimensional short memory parameter

Which parameters are of interest?

When considering *long memory* processes, we are usually primarily interested in the parameter d (and possibly μ). The parameters σ, ϕ, θ (and even p, q) are essentially nuisance parameters.

ARFIMA parameters

- μ – location parameter
- σ – scale parameter
- d – long memory parameter (long memory process iff $0 < d < 0.5$)
- ϕ – p-dimensional short memory parameter
- θ – q-dimensional short memory parameter

Which parameters are of interest?

When considering *long memory* processes, we are usually primarily interested in the parameter d (and possibly μ). The parameters σ, ϕ, θ (and even p, q) are essentially nuisance parameters.

ARFIMA parameters

- μ – location parameter
- σ – scale parameter
- d – long memory parameter (long memory process iff $0 < d < 0.5$)
- ϕ – p-dimensional short memory parameter
- $\boldsymbol{\theta}$ – q-dimensional short memory parameter

Which parameters are of interest?

When considering *long memory* processes, we are usually primarily interested in the parameter d (and possibly μ). The parameters $\sigma, \phi, \boldsymbol{\theta}$ (and even p, q) are essentially nuisance parameters.

ARFIMA parameters

- μ – location parameter
- σ – scale parameter
- d – long memory parameter (long memory process iff $0 < d < 0.5$)
- ϕ – p-dimensional short memory parameter
- $\boldsymbol{\theta}$ – q-dimensional short memory parameter

Which parameters are of interest?

When considering *long memory* processes, we are usually primarily interested in the parameter d (and possibly μ). The parameters $\sigma, \phi, \boldsymbol{\theta}$ (and even p, q) are essentially nuisance parameters.

Outline

Outline of problem

- Assume Gaussian distribution for the innovations
- Bayesian: use flat priors for μ, $\log(\sigma)$, and d...

Outline of problem

- Assume Gaussian distribution for the innovations
- Bayesian: use flat priors for μ, $\log(\sigma)$, and d...
- ... but can use any set of (independent) priors if desired.

Outline of problem

- Assume Gaussian distribution for the innovations
- Bayesian: use flat priors for μ, $\log(\sigma)$, and d...
- ... but can use any set of (independent) priors if desired.

- Even assuming Gaussianity, the likelihood for d is very complex – impossible to find analytic posterior

Outline of problem

- Assume Gaussian distribution for the innovations
- Bayesian: use flat priors for μ, $\log(\sigma)$, and d...
- ... but can use any set of (independent) priors if desired.

- Even assuming Gaussianity, the likelihood for d is very complex – impossible to find analytic posterior
- Must resort to MCMC methods in order to obtain samples from the posterior

Outline of problem

- Assume Gaussian distribution for the innovations
- Bayesian: use flat priors for μ, $\log(\sigma)$, and d...
- ... but can use any set of (independent) priors if desired.

- Even assuming Gaussianity, the likelihood for d is very complex – impossible to find analytic posterior
- Must resort to MCMC methods in order to obtain samples from the posterior
- Don't want to *assume* form of short memory (i.e. p, q) – must use Reversible-Jump (RJ) MCMC [Green, 1995]

Outline of method

- Re-parameterisation of model to enforce stationarity constraints on Φ and Θ

Outline of method

- Re-parameterisation of model to enforce stationarity constraints on Φ and Θ
- Efficient calculation of Gaussian likelihood (long memory correlation structure prevents use of standard quick methods)

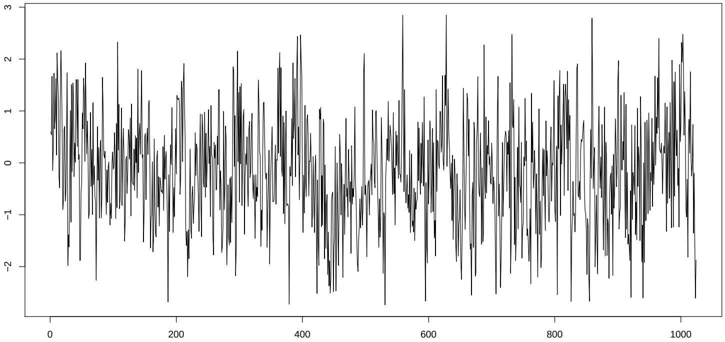
Outline of method

- Re-parameterisation of model to enforce stationarity constraints on Φ and Θ
- Efficient calculation of Gaussian likelihood (long memory correlation structure prevents use of standard quick methods)
- Necessary use of Metropolis–Hastings algorithm requires careful selection of proposal distributions

Outline of method

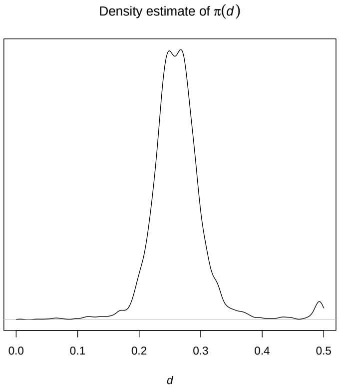
- Re-parameterisation of model to enforce stationarity constraints on Φ and Θ
- Efficient calculation of Gaussian likelihood (long memory correlation structure prevents use of standard quick methods)
- Necessary use of Metropolis–Hastings algorithm requires careful selection of proposal distributions
- Correlation between parameters (e.g. ϕ and d) requires blocking.

Example: ‘Pure’ Gaussian Long Range Dependence

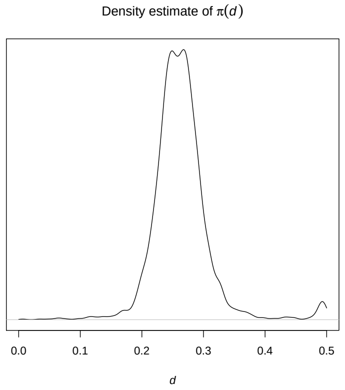


$$(1-B)^{0.25}X_t = \varepsilon_t$$

Example: ‘Pure’ Gaussian Long Range Dependence

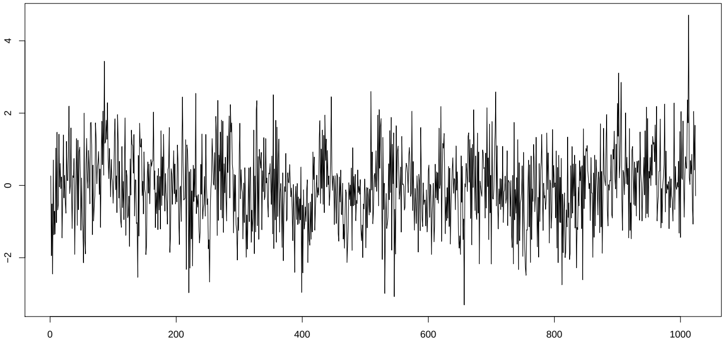


Example: ‘Pure’ Gaussian Long Range Dependence



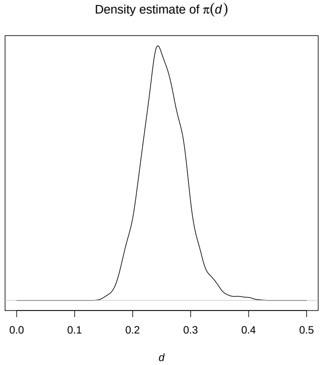
- Similarly good results for μ and σ
- The posterior model probability for the $(0, d, 0)$ model was 70%

Example: ‘Corrupted’ Gaussian Long Range Dependence

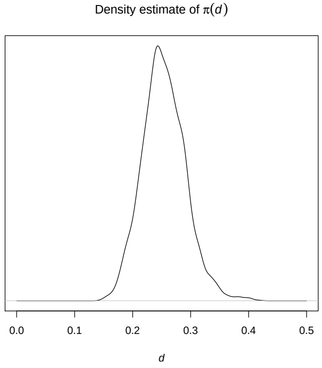


$$(1 + 0.75B)(1 - B)^{0.25} X_t = (1 + 0.5B)\varepsilon_t$$

Example: 'Corrupted' Gaussian Long Range Dependence



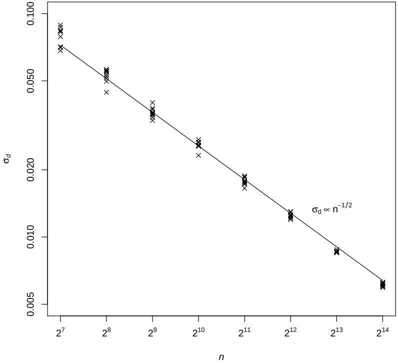
Example: 'Corrupted' Gaussian Long Range Dependence



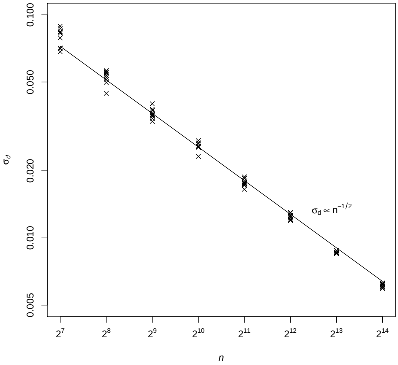
- The posterior model probability for the $(1, d, 1)$ model was 77%
- The posterior model probability for the $(0, d, 0)$ model was 0%

Dependence of posterior variance on n

Dependence of posterior variance on n



Dependence of posterior variance on n



$$\sigma_d \propto n^{-1/2}$$

Outline

Assumptions and general method

- Drop the Gaussianity assumption
- Replace with a more general distribution (e.g. α-stable)
- Seek *joint* inference about d and α

Assumptions and general method

- Drop the Gaussianity assumption
- Replace with a more general distribution (e.g. α-stable)
- Seek *joint* inference about d and α

- Initially (for simplicity) we assume no short memory, i.e. we assume a $(0, d, 0)$ model

Assumptions and general method

- Drop the Gaussianity assumption
- Replace with a more general distribution (e.g. α-stable)
- Seek *joint* inference about d and α

- Initially (for simplicity) we assume no short memory, i.e. we assume a $(0, d, 0)$ model
- Infinite variance means that auto-covariance approach is no longer sound

Assumptions and general method

- Drop the Gaussianity assumption
- Replace with a more general distribution (e.g. α-stable)
- Seek *joint* inference about d and α

- Initially (for simplicity) we assume no short memory, i.e. we assume a $(0, d, 0)$ model
- Infinite variance means that auto-covariance approach is no longer sound
- Lack of closed form for α-stable density implies lack of closed form for likelihood

Solution

- Approximate the long memory process as a very high order AR process

Solution

- Approximate the long memory process as a very high order AR process
- Construct the likelihood sequentially and evaluate using specialised efficient methods

Solution

- Approximate the long memory process as a very high order AR process
- Construct the likelihood sequentially and evaluate using specialised efficient methods

$$f(x_1, \ldots, x_t|\mathcal{H}) = f(x_t|x_{t-1}, \ldots, x_1, \mathcal{H})f(x_{t-1}, \ldots, x_1|\mathcal{H})$$

where $\mathcal{H}$ is the finite recent history of the process $x_0, x_{-1}, \ldots, x_{-n}$

Solution

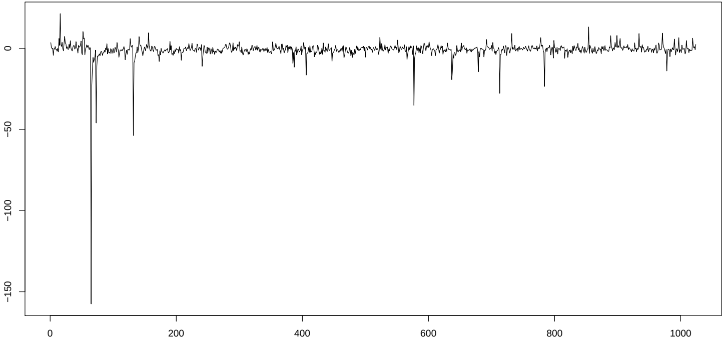
- Approximate the long memory process as a very high order AR process
- Construct the likelihood sequentially and evaluate using specialised efficient methods

$$f(x_1, \ldots, x_t|\mathcal{H}) = f(x_t|x_{t-1}, \ldots, x_1, \mathcal{H})f(x_{t-1}, \ldots, x_1|\mathcal{H})$$

where $\mathcal{H}$ is the finite recent history of the process $x_0, x_{-1}, \ldots, x_{-n}$

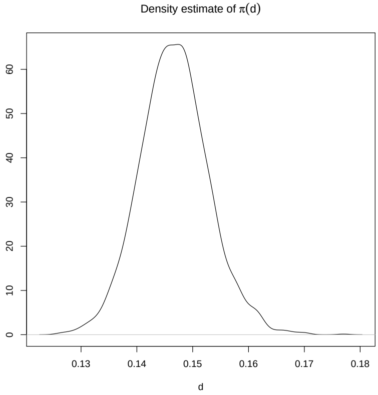
- Use auxiliary variables to integrate out the (unknown) history $\mathcal{H}$
- In practice, setting $\mathcal{H} = \bar{x}, \ldots, \bar{x}$ suffices, providing enormous computational saving.

Example: 'Pure' symmetric α-stable long memory

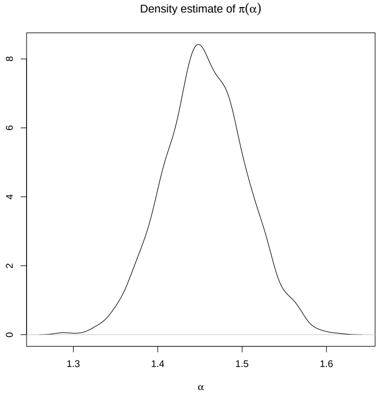


$$(1-B)^{0.15}X_t = \varepsilon_t, \qquad \alpha = 1.5$$

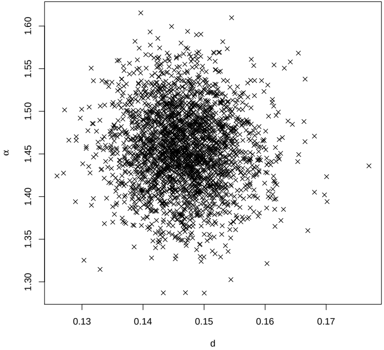
Example: 'Pure' symmetric α-stable long memory



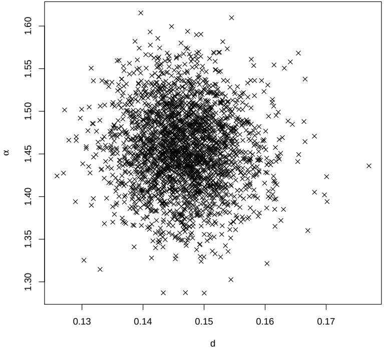
Example: 'Pure' symmetric α-stable long memory



Example: 'Pure' symmetric α-stable long memory



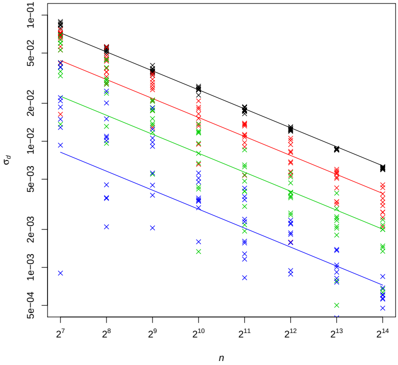
Example: 'Pure' symmetric α-stable long memory



- Good estimation of all parameters
- The posteriors of d and α are independent

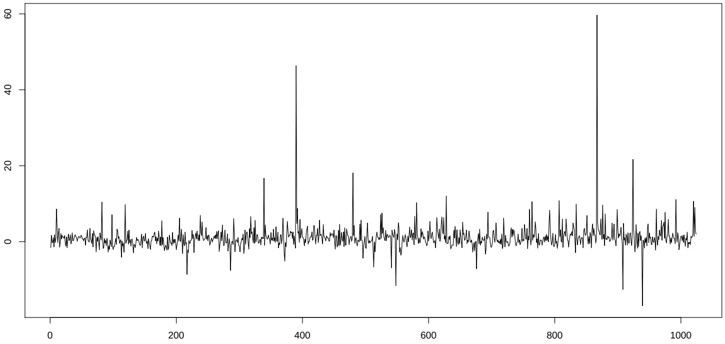
Dependence of posterior variance on n

Dependence of posterior variance on n



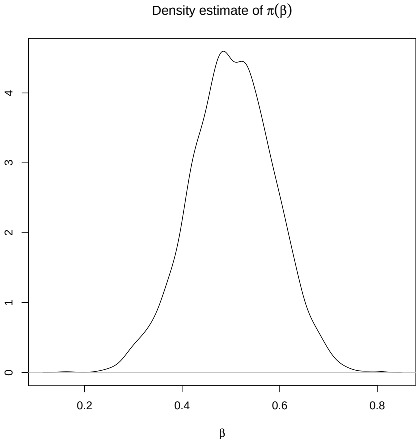
$$\sigma_d \propto n^{-1/2}$$

Example: ‘Pure’ asymmetric α-stable long memory

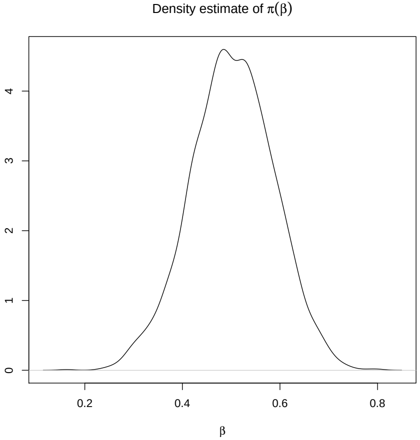


$$(1 - B)^{0.1} X_t = \varepsilon_t, \qquad \alpha = 1.5 \qquad \beta = 0.5$$

Example: ‘Pure’ asymmetric α-stable long memory



Example: 'Pure' asymmetric α-stable long memory



- Good estimation of all other parameters

References

Green, P. J. (1995).
Reversible jump Markov chain Monte Carlo computation and Bayesian model determination.
Biometrika, 82(4), 711–732.

Samorodnitsky, G. (2006).
Long range dependence.
Foundations and Trends in Stochastic Systems, 1(3), 163–257.