Proceedings of the Workshop

Inference and Estimation in Probabilistic Time-Series Models

18 June to 20 June 2008

Isaac Newton Institute for Mathematical Sciences, Cambridge, UK

Workshop Organisers:
David Barber, Ali Taylan Cemgil, Silvia Chiappa

Table of Contents

Bayesian study of Stochastic volatility models with STAR volatilities and Leverage effect.

Esmail Amiri*
Member of faculty at IKIU International University
Departmrnt of Statistics Imam Khomeini International University
Ghazvin, Iran.
e_amiri@yahoo.com, e_amiri@ikiu.ac.ir

Abstract

The results of time series studies present that a sequence of returns on some financial assets often exhibit time dependent variances and excess kurtosis in the marginal distributions. Two kinds of models have been suggested by researchers to predict the returns in this situation: observation-driven and parameter driven models. In parameter-driven models, it is assumed that the time dependent variances are random variables generated by an underlying stochastic process. These models are named stochastic volatility models(SV). In a Bayesian frame work we assume the time dependent variances follow a non-linear autoregressive model known as *smooth transition autoregressive(STAR)* model and also leverage effect between volatility and mean innovations is present. To estimate the parameters of the SV model, Markov chain Monte Carlo(MCMC) methods is applied. A data set of log transformed Pound/Dollar exchange rate is analyzed with the proposed method. The result showed that SV-STAR performed better than SV-AR.

keywords : Stochastic volatility, Smooth transition autoregressive, Markov chain Monte Carlo methods, Bayesian , Deviance information criterion, Leverage effect.

1 Introduction

There is overwhelming evidence in study of financial time series that a sequence of returns $\{y_t\}$ on some financial assets such as stocks cannot be modeled by the linear models, because of time dependent variances and excess kurtosis in the marginal distributions.

Based on time dependent variances two classes of models have been suggested by researchers, namely GARCH(Generalized Autoregressive Conditional Heteroskedasticity) and SV(Stochastic Volatility). Both of these models estimate volatility conditional on past information and are not necessarily direct competitors but rather the complements of each other in certain respects.

The class of GARCH models, builds on the fact that the volatility is time varying and persistent and, also current volatility depends deterministically on past volatility and the past squared returns. GARCH models are easy to estimate and quite popular since it is relatively straight forward to evaluate the likelihood function for this kind of models. A standard $GARCH(1,1)$, for instance, takes the following form to explain the variance h_t at time t:

$$\begin{aligned} y_t &= \sqrt{h_t}\epsilon_t \\ h_t &= \beta_0 + \beta_1 y_{t-1}^2 + \beta_2 h_{t-1} \end{aligned} \tag{1}$$

where y_t is the return on an asset at time $t = 1, ..., T$. $\{\epsilon_t\}$ is independent Gaussian white noise processes. Given the observation up to time $t-1$, the volatility h_t at time t is deterministic, once the parameters $(\beta_0, \beta_1, \beta_2)$ are known, Bollerslev(1986). For the class of SV models, the innovations to

*http://www.ikiu.ac.ir.

the volatility are random and the volatility realizations are therefore unobservable and more difficult to be covered from data. However, it is impossible to write the likelihood function of SV models in a simple closed form expression. Estimating an SV model involves integrating out the hidden volatilities.

In the literature to estimate SV models there are several methods, one method is MCMC.

Markov Chain Monte Carlo methods(MCMC) is, a promising way of attacking likelihood estimation by simulation techniques using the computer intensive Markov Chain Monte Carlo methods to draw samples from the distribution of volatilities conditional on observations. Kim and Shephard(1994) and Jacuier et al.(1994) are among the first pioneers who applied MCMC methods to estimate SV models.

The aim is to inference on a class of stochastic volatiliy models known as *Stochastic volatility with smooth transition autoregressive*(SV-STAR) in a Bayesain framework via MCMC , as in Jacquier et al.(1994, 1999), while assuming leverage effect between volatility and mean innovations is present. MCMC permits to obtain the posterior distribution of the parameters by simulation rather than analytical methods.

In section 2 and 3 the class of Stochastic volatility with smooth transition is introduced, section 4 is devoted to MCMC methods, in section 5 deviance information criterion and in section 6 conditional posterior distributions is presented, in section 7 an algorithm is proposed, in the two final sections an application is displayed and an illustrating discussion is presented.

2 Smooth transition autoregressive models(STAR)

A popular class of non-linear time series models is the threshold autoregressive models(TAR), which is probably first proposed by Tong(1978). A TAR model is a piece-wise linear model which is reach enough to generate complex non-linear dynamics. These models are suitable to model periodic time series, or produce asymmetric and jump phenomena that can not be captured by linear time series models, Ziwot and Wang(2006). Let observation at time t is denoted by λ_t, then a TAR model with $k-1$ threshold values can be presented as follows:

$$\lambda_t = X_t\phi^{(j)} + \sigma^{(j)}\eta_t \qquad if \quad r_{j-1} < z_t \leq r_j \tag{2}$$

where $X_t = (1, \lambda_{t-1}, \lambda_{t-2}, \cdots, \lambda_{t-p})$, $j = 1, 2, \cdots, k$, $-\infty = r_0 < r_1 < \cdots < r_k = \infty$, $\eta \sim N(0,1)$, $\phi^{(j)} = (1, \phi_1^{(j)}, \phi_2^{(j)}, \cdots, \phi_p^{(j)})$, z_t is the threshold variable and $r_1, r_2, \ldots, r_{k-1}$ are the threshold values. These values divide the domain of the threshold variable z_t into k different regimes. In each different regime, the time series λ_t follows a different $AR(p)$ model. When the threshold variable $z_t = \lambda_t$, with the delay parameter d being a positive integer, the regimes of λ_t is determined by its own laged value λ_{t-d} and the TAR model is called *self exiting* TAR or SETAR model.

In the TAR models, a regime switch happens when the threshold variable crosses a certain threshold. In some cases it is reasonable to assume that the regime switch happens gradually in a smooth fashion. If the discountinuity of the threshold is replaced by a smooth transition function, TAR models can be generalized to *smooth transition autoregressive* (STAR) models. Two main (STAR) models are *logistic* and *exponential*.

2.1 Logistic and Exponential STAR models

In a two regime SETAR model, the observations λ_t are generated either from the first regime when λ_{t-d} is smaller than the threshold, or from the second regime when λ_{t-d} is greater than the threshold value. If the binary indicator function is replaced by a smooth transition function $0 < F(z_t) < 1$ which depends on a transition variable z_t (like the threshold variable in TAR models), the model is called smooth transition autoregressive (STAR) model. A general form of STAR model is as follows,

$$\lambda_t = X_t\phi^{(1)}(1 - F(z_t)) + X_t\psi(F(z_t)) + \eta_t \qquad \eta_t \sim N(0, \sigma^2) \tag{3}$$

where $\psi = (1, \psi_1, \cdots, \psi_p)$. For practical computation, let $\phi^{(2)} = \psi - \phi^{(1)}$, then equation (3)can be rewritten as

$$\lambda_t = X_t\phi^{(1)} + X_t\phi^{(2)}(F(z_t)) + \eta_t \tag{4}$$

Model (4) is similar to a two regime SETAR model. Now the observations λ_t switch between two regimes smoothly in the sense that the dynamics of λ_t my be determined by both regimes, with one regime having more impacts in sometimes and the other regime having more impacts in other times.

Two popular choices for the smooth transition function are the *logistic function* and the *exponential function* as follows, respectively.

$$F(z_t, \gamma, c)) = [1 + e^{-\gamma(z_t - c)}]^{-1}, \qquad \gamma > 0 \tag{5}$$

$$F(z_t, \gamma, c)) = 1 - e^{-\gamma(z_t - c)^2}, \qquad \gamma > 0 \tag{6}$$

the resulting model is referred to as logistic STAR or LSTAR model and exponential STAR or ESTAR, respectively. In the equations (5)and (6) the parameter c is interpreted as the threshold as in TAR models, and γ determines the speed and smoothness of the transition.

3 Stochastic volatility models with STAR volatilities

The following lognormal SV model is well known in the stochastic volatility litreature(e.g, Harvey and Shephard (1996)),

$$\begin{aligned} y_t &= \sqrt{h_t}\epsilon_t \\ \log h_{t+1} &= \alpha + \delta \log h_t + \sigma_\eta \eta_{t+1} \end{aligned} \tag{7}$$

where y_t is the return on an asset at time $t = 1, ..., T$. $\{\epsilon_t\}$ and $\{\eta_t\}$ are independent Gaussian white noise processes, σ_η is the standard deviation of the shock to $\log h_t$ and $\log h_t$ has a normal distribution. We take the approach of Yu(2005) and assume $corr(\epsilon_t, \eta_{t+1}) = \rho$, then the covariance matrix of vector $(\epsilon_t, \eta_{t+1})'$ is Ω,

$$\Omega = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix} \tag{8}$$

the parameter ρ measures the leverage effect. The leverage effect refers to the negative correlation between$\{\epsilon_t\}$ and $\{\eta_{t+1}\}$(eg. Yu(2005)), which could be the result of increase in volatility following a drop in equity returns.

Different models have been proposed for generating the volatility sequence h_t in the literature, (Kim, Shephard, & Chib (1998)).

Our aim is in a Bayesian approach to allow the volatility sequence to evolve according to the equation of a STAR(p) model as model (4), and also assume the leverage effect is present in the model. Then we name the SV model, stochastic volatility model with STAR volatilities (SV-STAR) and leverage effect. The equation of a SV-STAR with leverage effect model is as follows,

$$\begin{aligned} y_t &= \sqrt{h_t}\epsilon_t \qquad \epsilon_t \sim N(0,1) \qquad \eta_t \sim N(0,1) \\ \lambda_{t+1} &= X_t\phi^{(1)} + X_t\phi^{(2)}(F(\gamma, c, \lambda_{t-d})) + \sigma\eta_{t+1} \end{aligned} \tag{9}$$

where $\lambda_t = \log h_t$, $\phi^{(1)}$ and $\phi^{(2)}$ are $p+1$ dimensional vectors,
$corr(\epsilon_t, \eta_{t+1}) = \rho$, and $F(\gamma, c, \lambda_{t-d})$ is a smooth transition function. We assume, without loss of generality that, $d \leq p$ always. When $p = 1$, the STAR(1) reduces to an AR(1) model. In $F(\gamma, c, \lambda_{t-d})$, $\gamma > 0$, c and d are smoothness, location (threshold) and delay parameters, respectively. When $\gamma \to \infty$, the STAR model reduces to a SETAR model, and when $\gamma \to 0$, the standard AR(p) model arises. We assume that $\lambda_{-p+1}, \lambda_{-p+2}, \cdots, \lambda_0$ are not known quantities.

For the sake of computational purposes, the second equation of the (9) is presented in a matrix form,

$$\lambda_{t+1} = W'\theta + \sigma\eta_{t+1} \tag{10}$$

where $\theta' = (\phi^{(1)}, \phi^{(2)})$ and $W' = (X_t, X_t F(\gamma, c, \lambda_{t-d}))$. Also let $\Theta = (\theta, \gamma, c, \sigma^2, \rho)$. Then we rewrite (9) as follows:

$$\begin{aligned} y_t &= e^{\lambda t/2}\epsilon_t \qquad \epsilon_t \sim N(0,1) \qquad \eta_t \sim N(0,1) \\ \lambda_{t+1} &= W'\theta + \sigma\eta_{t+1} \end{aligned} \tag{11}$$

where

$$\begin{pmatrix} \epsilon_t \\ \eta_{t+1} \end{pmatrix} \sim \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

4 Markov chain Monte Carlo methods (MCMC)

Markov chain Monte Carlo methods (MCMC) have virtually revolutionized the practice of Bayesian statistics. Early work on these methods pioneered by Hastings (1970) Geman and Geman (1984) while recent developments appears in Gelfand and smith (1990) and Chib and Greenberg (1996).

When sampling from some high-dimensional posterior densities are intractable, MCMC methods provide us with the algorithms to achieve the desired samples. Letting $\pi(\theta)$ be the interested target posterior distribution, the main idea behind MCMC is to build a Markov chain transition kernel

$$P(z, C) = Pr\{\theta^{(m)} \in C | \theta^{(m-1)} \in z\}_{m=1}^{M} \tag{12}$$

Starting from some initial state $\theta^{(0)}$, with limiting invariant distribution equal to $\pi(\theta)$. It has been proved that(see Chib and Greenberg (1996)) under some suitable conditions, one can build such a transition kernel generating a Markov chain $\{\theta^{(m)}|\theta^{(m-1)}\}$ whose realizations converge in distribution to $\pi(\theta)$). Once convergence is happened, a sample of serially dependent simulated observation on the parameter θ is obtained, which can be used to perform Monte Carlo inference. Much effort has been devoted to the design of algorithms able to generate a convergent transition kernel. The Metropolis-Hastings(MH) and the Gibbs sampler are the among most famous algorithms which are very effective in buildings the above mentioned Markov chain transition kernel.

5 The deviance information criteria

Following the original suggestion of Dempster(1974), recently a model selection criteria in the Bayesian framework is developed, Spiegelhalter et al.(2002). This criteria is named *Deviance Information Criterion*(DIC) which is a generalization of well known AIC(Akaike, information criterion). This criteria is preferred to, BIC(Bayesian information criterion) and AIC, because, unlike them, DIC needs to effective number of parameters of the model and applicable to complex hierarchical random effects models. DIC is defined based on the posterior distribution of the classical deviance $D(\Theta)$, as follows:

$$D(\Theta) = -2 \log f(y|\Theta) + 2 \log f(y) \tag{13}$$

where y and Θ are vectors of observations and parameters, respectively.

$$DIC = \bar{D} + p_D \tag{14}$$

$\bar{D} = E_{\Theta|y}[D]$ and $p_D = E_{\Theta|y}[D] - D(E_{\Theta|y}[\Theta]) = \bar{D} - D(\bar{\Theta})$. Also DIC can be presented as

$$DIC = \hat{D} + 2p_D \tag{15}$$

where $\hat{D} = D(\bar{\Theta})$

6 Conditional posterior distributions.

Equation (11) implies a bivariate normal for $y_t|\lambda_t, \rho$ and
$\lambda_{t+1}|\lambda_t, \theta, \gamma, c, \sigma^2, \rho$. By writing this bivariate normal density as the product of the density of $\lambda_{t+1}|\lambda_t, \theta, \gamma, c, \sigma^2$ and conditional density of $y_t|\lambda_{t+1}, \lambda_t, \theta, \gamma, c, \sigma^2, \rho$ it is easily seen that

$$\lambda_{t+1}|\lambda_t, \theta, \gamma, c, \sigma^2 \sim N(W'\theta, \sigma^2) \tag{16}$$

$$y_t|\lambda_{t+1}, \lambda_t, \theta, \gamma, c, \sigma^2, \rho \sim N[\frac{\rho}{\sigma} e^{\lambda_t/2}(\lambda_{t+1} - W'\theta), e^{\lambda_t}(1-\rho^2)] \tag{17}$$

assuming, the above conditional distributions are independent for $y_t, t = 1, \cdots, T$, therefore

$$f(y_1, y_2, \cdots, y_T|\Theta, \lambda) = \prod_{i=1}^{T} \frac{1}{(2\pi)^{1/2} e^{\lambda_t/2}(1-\rho^2)^{1/2}} e^{-\frac{1}{2e^{\lambda_t}(1-\rho^2)}[y_t - \frac{\rho}{\sigma} e^{\lambda_t/2}(\lambda_{t+1} - W'\theta)]^2} \tag{18}$$

then,

$$f(y_1, y_2, \cdots, y_T|\Theta, \lambda) = \frac{1}{(2\pi)^{T/2}(1-\rho^2)^{T/2}} e^{-\frac{1}{2(1-\rho^2)}\{\sum_{t=1}^{T}[y_t e^{-\lambda_t/2} - \frac{\rho}{\sigma}(\lambda_{t+1} - W'\theta)]^2 + (1-\rho^2)\lambda_t\}}. \tag{19}$$

Equation (19)is the likelihood.

Let assume p and d are known. Applying Lubreno's(2000) formulation, we assume the following priors ,

$$p(\gamma) = \frac{1}{1+\gamma^2}, \qquad \gamma > 0$$

where $p(\gamma)$ is a truncated cauchy density.

$$c \sim U[c_1, c_2]$$

where c has a uniform density, $c \in [c_1, c_2], c_1 = \hat{F}(0.15), c_2 = \hat{F}(0.85)$ and $\hat{F}$ is the emprical cumulative distribution function(cdf) of the time series.

$$p(\sigma^2) \propto \frac{1}{\sigma^2}$$

ρ is assumed to be uniformly distributed, $\rho \in (-1, 1)$.

With the assumption of independence of γ, c, σ^2, ρ and $\phi^{(1)}$ and also an improper prior for $\phi^{(1)}$,

$$p(\phi^{(1)}, \gamma, \sigma^2, c, \rho) \propto (1+\gamma^2)^{-1}\sigma^{-2}$$

$$(\phi^{(2)}|\sigma^2, \gamma, \rho) \sim N(0, \sigma^2 e^{\gamma} I_{p+1})$$

Then, the joint prior density is,

$$p(\Theta) \propto \sigma^{-3}(1+\gamma^2)^{-1}\exp\{-\frac{1}{2}(\gamma + \sigma^{-2}e^{-\gamma}\phi'^{(2)}\phi^{(2)})\} \tag{20}$$

A full Bayesian model consists of the joint prior distribution of all unknown parameters, here, Θ, and the unknown states, $\lambda = (\lambda_{-p+1}, \cdots, \lambda_0, \lambda_1, \cdots, \lambda_T)$, and the likelihood. Bayesian inference is then based on the posterior distribution of the unknowns given the data. By successive conditioning, the prior density is

$$\begin{aligned} p(\Theta, \lambda) \quad &= p(\Theta)p(\lambda_0, \lambda_{-1}, \cdots, \lambda_{-p+1}|\sigma^2, \rho)\times \\ &\prod_{t=1}^{T} p(\lambda_t|\lambda_{t-1}, \cdots, \lambda_{t-p}, \Theta) \end{aligned} \tag{21}$$

where, we assume

$$(\lambda_0, \lambda_{-1}, \cdots, \lambda_{-p+1}|\sigma^2) \sim N(0, \sigma^2 I_p)$$

and

$$(\lambda_{t+1}|\lambda_t, \lambda_{t-1}, \cdots, \lambda_{t-p+1}, \Theta) \sim N(W'\theta, \sigma^2)$$

Therefore

$$\begin{aligned} p(\Theta, \lambda) \propto\ & \sigma^{-(T+3+p)}(1+\gamma^2)^{-1} \\ & e^{-\frac{1}{2\sigma^2}\{(\sigma^2\gamma + e^{-\gamma}\phi'^{(2)}\phi^{(2)}) + \sum_{t=0}^{-p+1}\lambda_t^2 + \sum_{t=1}^{T}(\lambda_{t+1} - W'\theta)^2\}} \end{aligned} \tag{22}$$

Using the Bayes theorem, the joint posterior distribution of the unknowns given the data is proportional to the prior times the likelihood, i.e,

$$\begin{aligned} \pi(\Theta, \lambda|y_1, \cdots, y_T) \propto\ & (1+\gamma^2)^{-1}\sigma^{-(T+p+3)}(1-p^2)^{-T/2}\times \\ & \exp\{-\tfrac{1}{2\sigma^2}[\sigma^2\gamma + e^{-\gamma}\phi'^{(2)}\phi^{(2)} + \sum_{t=0}^{-p+1}\lambda_t^2 + \sum_{t=1}^{T}[(\lambda_t - W'\theta)^2]] \\ & -\tfrac{1}{2(1-\rho^2)}\{\sum_{t=1}^{T}[y_t e^{-\lambda_t/2} - \tfrac{\rho}{\sigma}(\lambda_{t+1} - W'\theta)]^2 + (1-\rho^2)\lambda_t\}\} \end{aligned} \tag{23}$$

In order to apply MCMC methods, full conditional distributions are necessary, the full conditionals are as follows:

$$\pi(\Theta|\lambda) \propto \frac{\sigma^{-(T+6)/2}}{(1+\gamma^2)}\exp\{-\frac{1}{\sigma^2}[\gamma\sigma^2 + e^{-\gamma}\phi'^{(2)}\phi^{(2)} + \sum_{t=1}^{T}(\lambda_t - W_t'\theta)^2]\} \tag{24}$$

$$\lambda_t|\lambda_{-t} \sim N(W'\theta, \sigma^2), \quad \lambda_{-t} = (\lambda_{-p+1}, \cdots, \lambda_0, \lambda_1, \cdots, \lambda_{t-1}, \lambda_{t+1}, \cdots, \lambda_T) \tag{25}$$

$$(\theta|\lambda_t, \gamma, c) \sim N\{[\sum W_t W_t' \sigma^{-2} + M](\sum W_t \lambda_t \sigma^{-2}), (\sum W_t W_t' \sigma^{-2} + M)\} \tag{26}$$

where $M = diag(0, \sigma^2 e^{-\gamma} I_{p+1})$.

$$(\sigma^2|\lambda, \Theta) \sim IG[\frac{T+p+1}{2}, (e^{\gamma}\phi'^{(2)}\phi^{(2)} + \sum(\lambda_t - W'\theta)^2)/2] \tag{27}$$

where IG denotes inverse gamma density function.

$$f(\gamma, c|\lambda, \theta) \propto \frac{\sigma^{-(T+6)/2}}{1+\gamma^2} \exp\{-\frac{1}{2\sigma^2}[\gamma\sigma^2 + e^{-\gamma}\phi'^{(2)}\phi^{(2)} + \sum_{t=1}^{T}(\lambda_t - W'\theta)^2]\} \tag{28}$$

$$\begin{aligned} f(\lambda_t|\lambda_{-t}, \Theta, y) &\propto f(y_t|\lambda_t) \prod_{i=0}^{p} f(\lambda_{t+i}|\lambda_{t+i-1}, \cdots, \lambda_{t+i-p}; \Theta) \\ &= g(\lambda_t|\lambda_{-t}, \Theta, y) \end{aligned} \tag{29}$$

If p and d are not known, their conditional posterior distributions can be calculated as follows.

Let $p(d)$ be the prior probability of the $d \in \{1, 2, \cdots, L\}$, where L is a known positive integer. Therefore the conditional posterior distribution of d is

$$\pi(d|\lambda, \theta) \propto f(d|\lambda, \theta)p(d) \propto \frac{\sigma^{-(T)/2}}{(2\pi)^{T/2}} \exp\{-\frac{1}{\sigma^2}\sum_{t=1}^{T}(\lambda_t - W_t'\theta)^2\} \tag{30}$$

Let $p(p)$ be the prior probability of the $p \in \{1, 2, \cdots, N\}$, where N is a known positive integer, multiplying the prior by the likelihood and integrating out the θ, the conditional posterior distribution of p is

$$\begin{aligned} \pi(k|\lambda, \gamma, c, d, \sigma^2) \quad &\propto (2\pi)^{(k+1)/2}[\sigma^2 e^{\gamma}]^{-\frac{k+1}{2}} |\sum_{t=1}^{T} W_t W_t' \sigma^{-2} + M|^{1/2} \\ &\exp\{-\tfrac{1}{2}(\sigma^{-2}\lambda'\lambda - [\sum_{t=1}^{T} W_t \lambda_t \sigma^{-2}] \\ &[\sum_{t=1}^{T} W_t W_t' \sigma^{-2} + M]'[\sum_{t=1}^{T} W_t W_t' \sigma^{-2} + M]^{-1} \\ &[\sum_{t=1}^{T} W_t W_t' \sigma^{-2} + M][\sum_{t=1}^{T} W_t \lambda_t' \sigma^{-2}])\} \end{aligned} \tag{31}$$

7 Algorithm

In our application y, $\lambda$and Θ are the vector of observation, the vector of log volatilities and the vector of identified unknown parameters, respectively. following kim et al.(1998)
$\pi(y|\Theta) = \int \pi(y|\lambda, \Theta)\pi(\lambda|\Theta)d\lambda$ is the likelihood function, the calculation of this likelihood function is intractable.

The aim is to sample the augmented posterior density $(\lambda, \Theta|y)$ that includes the latent volatilities λ as unknown parameters.

To sample the posterior density $\pi(\lambda, \Theta|y)$ following Jacquier et al.(1994) full conditional distribution of each component of $\pi(\lambda, \Theta|y)$is necessary. The sampling strategy when p and d are known is as follows

1. Initialize the volatilities and the parameter vector at some $\lambda^{(0)}$ and $\Theta^{(0)}$ respectively.
2. Simulate the volatility vector λ^i from the following full conditional
   $f(\lambda_t|\lambda_{-p+1}^{(i)}, \cdots, \lambda_1^{(i)}, \cdots, \lambda_{t-1}^{(i)}, \lambda_{t+1}^{(i-1)}, \cdots, \cdots, \lambda_T^{(i-1)}, \Theta^{(i-1)}, y)$
3. Sample θ from $(\theta|\lambda^{(i+1)}, \gamma^{(i)}, c^{(i)}, \sigma^{2^{(i)}})$
4. Sample σ^2 from $(\sigma^2|\lambda^{(i+1)}, \theta^{(i+1)})$

5. Sample γ and c from $f(\gamma, c|\lambda^{(i+1)}, \theta^{(i+1)})$ using MH algorithm.

6. If $i \leq m$ go to 2.

where m is the required number of iterations to generate samples from $\pi(\lambda, \Theta|y)$.

If p and d are not known, the following steps could be inserted before the algorithm's final step.

6. Sample d from $\pi(d|\lambda^{(i+1)}, \theta^{(i+1)})$

7. Sample k from $\pi(k|\lambda^{(i+1)}, \gamma^{(i+1)}, c^{(i+1)}, d^{(i+1)})$ using MH algorithm.

8 Application

We apply the method and estimation technique described above to a financial time series. The data consist of a time series of the daily Pound/Dollar exchange rates from 01/10/1981 to 28/6/1985. This data set has been previously studied by Harvey et al.(1994)and other authors. The series is daily log transformed, mean corrected returns $\{y_t\}$ given by the transformation

$$y_t = \log x_t - \log x_{t-1} - \frac{1}{T}\sum_{t=1}^{T}(\log x_t - \log x_{t-1}), \quad t = 1, \cdots, T \tag{32}$$

where $\{x_t\}$ is daily exchange rates.

Ox and BRugs softwares is used to facilitate programming of simulation. In the examples smooth transition function is logistic function, but the exponential function can be easily replaced. Also to ease the comparison of our results with the results in the literature, see (Meyer and Yu(2000)), the parameters of the following form of AR(p) model in each regime is estimated

$$\lambda_t = \mu + \sum_{i=1}^{p}\phi_i(\lambda_{t-i} - \mu) + \eta_t, \qquad \eta_t \sim N(0, \sigma^2) \tag{33}$$

For the convergence control, as a rule of thumb the Monte Carlo error(MC-error) for each parameter of interest should be less than $5\%$of the sample standard deviation. Unfortunately because of page limitation, we are unable to present all of the results, therefore we present here only the final simulation results.

Parameters of a SV-AR(1) model is estimated, the result is as follows:

Table 1: DIC criterion for SV-AR(1).

	$\bar{D}$	$\hat{D}$	DIC	pD
y	1756	1706	1805	49.35
total	1756	1706	1805	49.35

Table 2:Estimated parameters for model SV-AR(1), $\beta = \exp(\mu/2)$.

par.	mean	sd	MC-error	2.5pc	median	97.5pc	start	sample
β	0.6983	0.10490	0.0041120	0.5469	0.6789	0.95430	4002	29998
μ	-0.7390	0.28230	0.0110400	-1.2070	-0.7746	-0.09345	4002	29998
ϕ	0.9775	0.01117	0.0004986	0.9509	0.9790	0.99470	4002	29998
σ	0.1617	0.03018	0.0018200	0.1108	0.1574	0.23010	4002	29998
ρ	-.2017	0.05018	0.0018200	-0.1908	-0.1874	-0.11010	4002	29998

Parameters of a SV-STAR(1) with $d = 2$ is estimated the result is as follows:

Table 3: Dic criterion for a SV-STAR(1) with $d = 2$.

	$\bar{D}$	$\hat{D}$	DIC	pD
y	1745	1695	1795	49.61
total	1745	1695	1795	49.61

Table 4: Estimated parameters of a SV-STAR(1) with $d = 2$.

Par.	mean	sd	MC-error	2.5pc	median	97.5pc	start	sample
c	0.26610	0.29830	0.0115400	-0.50570	0.36640	0.5825	4002	39998
γ	15.23000	4.43500	0.0831300	7.00000	15.00000	24.0000	4002	39998
μ_1	0.04261	0.53400	0.0255300	-0.96540	0.036340	1.1030	4002	39998
μ_2	0.07591	0.05138	0.0024440	-0.00853	0.070130	0.1920	4002	39998
ϕ_1	0.98080	0.01333	0.0005524	0.94730	0.983300	0.9985	4002	39998
ϕ_2	0.01028	0.05222	0.0025420	-0.08373	0.007779	0.1274	4002	39998
σ	0.17140	0.03654	0.0020990	0.11650	0.165400	0.2514	4002	39998
ρ	-0.2300	0.04350	0.002313	-0.26000	-0.2000	-0.15000	4002	39998

9 Discussion

A SV model is comprised of two equations, the first equation is called observation and the second one is named state. In the literature linear and nonlinear equations are proposed for the state equation. In a Bayesian approach a nonlinear model called smooth transition autoregressive(STAR)is used as state equation. Then the new SV model is named SV-STAR, also the leverage effect between conditional volatility mean and return is assumed. To estimate parameters of SV-STAR with leverage effect model, likelihood is constructed. The likelihood is intractable and parameter estimation is performed using MCMC methods. A financial data set is examined. Applying DIC criterion, the result of examination shows that the SV-STAR with leverage effect models perform better than traditional SV-AR with leverage effect model for this data set. Assuming parameters p and d are unknown the convergence was very slow. For the future work in this context, we propose three directions:

1. Investigating new simulation algorithms to make convergence of samplers faster.
2. Assuming different variance of the error term in each regime.
3. Assuming the change between two regimes is made via a step transition function.

References

[1] Bollerslev, T. (1986). *Generalized Autoregressive Conditional Hetroskedasticity*. Journal of Econometrics,**31**, 307–327.

[2] Chib, S. and Greenberg, E. (1995). *Understanding the Metropolis-Hastings algorithm.* The American statistician **49**, 327–335.

[3] Chib, S. and Greenberg, E. (1996). *Markov Chain Monte Carlo simulation methods in, econometrics*. Econometrics Theory **12**, 409–431

[4] Dempster, A. P. (1974). *The direct use of likelihood for significance testing* Proceedings of Conference on Foundmental questions in Statistical Infernce, Department of theoretical Statistics: University of Aarhus, 335-352.

[5] Duffie, D., Singleton, K.J. (1993). Simulated moment estimation of Markov models of asset prices, Econometrica, **61**, 929–952.

[6] Gelfand, A. E. and Smith, A. F. M.(1990). Sampling-based approaches to calculating marginal densities, J. Amer. Statist. Assoc, **85** 398–409.

[7] Geman, S. and Geman, D. , (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. IEEE Trans. Pattn Anal. Mach. Intell., **6**, 721741.

[8] Harvey, A. C., Ruiz, E., Shepherd, N. , (1994). Multivariate stochastic volatility methods, Review of Economic studies,**61**,247–264.

[9] Hastings, W.K. (1970). *Monte Carlo sampling methods using Markov Chains and their applications* . biometrika **57**, 97–109.

[10] Jacquier E., Polson N. G. and Rossi P.(1994). *Bayesian analysis of stochastic volatility models*. Journal of Business & Econometric Statistics **12**, 371–417.

[11] Jacquier E., Polson N. G. and Rossi P.(1999). *Models and priors for multivariate stochastic volatility* . Working paper, CIRANO, forthcoming in Journal of Econometrics.

[12] Jacquier E., Polson N. G. and Rossi P.(2004). *Bayesian Analysis of Stochastic Volatility models with fat tails and Correlated errors* . Journal of Econometrics 122, 185212.

[13] Kim, S., shephard, N.G. and Chib, S.(1998). *Stochastic volatility models: conditional normality versus heavy-tailed distributions*. Review of Economic Studies **65**, 361–393.

[14]Lubarno, M. , (2000). Bayesian analysis of nonlinear time series models with a threshold, Proceedings of the eleventh international Symposium in Econometric theory. [15]Melino, A. and and Turnbull, SM. (1990). Pricing foreign currency options with stochastic volatility. J. Econometrics **45**, 239–265.

[16] Metropolis, N., Rosenbluth, M.N., Teller, A.H. and Teller, E. (1953) *Equations state calculations by fast computing machines*. Journal of Chemical physics **21**, 1087–1091.

[17] Meyer, R. and Yu, J. (2000).*BUGS for a Bayesian analysis of stochastic volatility models*. Econometrics Journal,**3**, 198–215. TNel98Nelson, D. B. , (1998). The time series behaviour of stock market volatility and returns, PhD thesis, MIT.

[18] Robert, R.P and casella, G. (1999). Monte Carlo statistical Methods, Springer.

[19] Ruiz, E. (1994). Quasi maximum likelihood estimation of stochastic volatility models, Journal of Econometrics, **63**, 289–306.

[20] Shephard, N.G. and Kim, S., (1994), "Comment of Bayesian analysis of stochastic volatility by Jacquier, Polson and Rossi", *Bussiness and Economics statistics*,**12**,4, 371–717.

[21]Steel,M.F.J. , (1998). Bayesian analysis of stochastic volatility models with flexible tails, Econometric Reviews **17**, 109–143.

[22] Spiegelhalter, D. J., Best, N. G., Carlin, B.P. and van,der Linde, A. (2002). *Bayesian measures of model complexity and fit*, Journal of the Royal Statistical Society, Series B,**64**, 583-639.

[1] Tierney, L. (1992). Markov Chains for exploring posterior distribution, Annals of Statistics **22**, 1701–1728.

[23] Tong, H. , (1978). On a threshold model, in Chen, C. H.(Ed.)Patern regognition and signal processing, Amsterdam, Sijhoff & Nordhoof

[24] Tsay, R. S. , (2002). Analysis of finantial Time Serie, Wiley Intersience: New York.

[25] Yu, Jun.(2005). *On leverage in a stochastic volatility model* Journal of Econometrics **127** 165-178.

[26] Zivot, E.,WANG Jia-hui ,(2006). Modeling Financial Time Series with S-Plus Springer-Verlag.

CaViaR via Bayesian Nonparametric Quantile Regression

Katerina Aristodemou[*]
Department of Mathematical Sciences
Brunel University
West London
katerina.aristodemou@brunel.ac.uk

Keming Yu
Department of Mathematical Sciences
Brunel University
West London
keming.yu@brunel.ac.uk

Abstract

The Conditional Autoregressive Value at Risk (CAViaR) model introduced by Engle and Manganelli (2004) is a very popular time series model for estimating the Value at Risk in finance. Value at Risk (VaR) is one of the most common measures of market risk and its measurement has many important applications in the field of risk management as well as for regulatory processes. In statistical terms, VaR is estimated by calculating a quantile of the distribution of the financial returns. Given a series of financial returns, Y, VaR is the value of y that satisfies $P(Y \leq y) = \theta$, for a given value of θ. Our aim in this paper is to demonstrate how non-parametric Bayesian quantile regression can be used for the inference and forecast of CAViaR by constructing a flexible dependence structure for the model and taking account of parameter uncertainty.

1 Introduction

Value at Risk (VaR) is one of the most common measures of market risk. Market risk is defined as the possibility of a decrease in the value of an investment due to movements in the market (Hull, 2000). The measurement of VaR has many important applications in the field of risk management and it is also equally useful for regulatory processes. An example for the latter is that Central Bank regulators use VaR to determine the capital that banks and other financial institutions are obliged to keep, in order to meet market risk requirements.

The calculation of VaR aims at representing the total risk in a portfolio of financial assets by using a single number. It is defined as the maximum possible loss, for a specific probability, in the value of a portfolio due to sudden changes in the market.

The investigation of different methodologies for the calculation of VaR is motivated by the distinct characteristic of financial data:

- Financial return distributions are leptokurtotic, i.e. they have heavier tails and a higher peak than a normal distribution.
- Equity returns are typically negatively skewed
- Squared returns have significant autocorrelation, this means that volatilities of market factors tend to cluster, i.e. the market volatilities are considered to be quasi-stable (stable in the short period but changing in the long run)

Several researchers have applied different methodologies by taking into consideration as many of the above factors as possible. All of the proposed models have a similar structure with the main differences relating to the way the distribution of the portfolio returns is estimated. The choice of the most appropriate methodology depends on the understanding of the assumptions underlying the data and on the comprehension of the mathematical models and the corresponding quantitative techniques (Manganelli and Engle, 2001).

[*] www.carisma.brunel.ac.uk.

This paper aims to demonstrate how non-parametric Bayesian quantile regression can be used for the inference and forecast of CAViaR by constructing a flexible dependence structure for the model and taking account of parameter uncertainty. The rest of the paper is structured as follows. In Section 2 we present the existing methodology for calculating VaR using Quantile Regression. In Section 3 we describe the CAViaR model as presented by Engle and Manganelli (2004). In Section 4 we give a brief introduction to Bayesian Quantile regression and in Sections 5 and 6 we present the proposed methodology, including Bayesian model setting and proper posterior discussion. In Section 7 we carry out some simulations and then in Section 8 we present a comparison of techniques using empirical data. The paper is finally concluded with a discussion.

2 Methodologies for the calculation of VaR

According to Yu *et al.* (2003), in statistical terms, VaR is estimated by calculating a quantile of the distribution of the financial returns. Given a series of financial returns, Y, VaR is the value of y that satisfies $P(Y \leq y) = \theta$, for a given value of θ. Formally, the calculation of VaR enables the following statement to be made "We are $(100-\theta)\%$ certain that we shall not loose more than y dollars in the next k days " (Chen and Chen, 2003).

The most common methodologies for the estimation of VaR can be separated into 3 categories: parametric models, semiparametric models and quantile regression approach.

Parametric models depend on the assumption that the log-returns follow a specific distribution and can be described by a GARCH framework (Giot and Laurent, 2004). GARCH models are designed to model the conditional heteroskedasticity in a time series of returns:

$$y_t = \mu_t + \varepsilon_t,$$
$$\varepsilon_t = \sigma_t z_t.$$

Consider the quantile regression model:

$$y_t = f(x_t; \omega) + \varepsilon_t, \tag{1}$$

and assume that the θ^{th} regression quantile of ε_t is the value, 0, for which $P(\boldsymbol{\varepsilon} < 0) = \theta$, instead of $E(\boldsymbol{\varepsilon}) = 0$ in mean regression. The θ^{th} quantile regression model of y_t given x_t, is then given by $q_\theta(y_t|x_t) = f(x_t; \omega)$. That is we assume that $\left(\int_{-\infty}^{0} f_\theta(\varepsilon) d\varepsilon = \theta \right)$, where $f(\bullet)$ denotes the error density. In classical quantile regression (Koenker and Hallock, 2001), the θ^{th} regression quantile of ε_t is the value of $\hat{\theta}$ that minimises the problem: $\sum_t \rho_\theta \left(y_t - f(x_t; \omega) \right)$, where ρ_θ is the loss function and is defined as:

$$\rho_\theta(u) = \theta \, u I_{[0,\infty)}(u) - (1-\theta) \, u I_{(-\infty,0)}(u), \tag{2}$$

where $I_{[a,b]}(u)$ is an indicator on $[a, b]$.

In conventional generalized linear models, the estimates of the unknown regression parameters are obtained by assuming that: 1) conditional on x_t, the random variables y_t are mutually independent with distributions $f(y_t; \mu_t)$ specified by the values of $\mu_t = E(y_t|x_t)$ and 2) for some known link function g, $g(\mu_t) = \mathbf{x}^T \boldsymbol{\beta}$.

3 CAViaR

Engle and Manganelli (2004) proposed an alternative, semi-parametric approach to VaR calculation, the Conditional Autoregressive Value at Risk (CAViaR) model. The CAViaR model is a very popular method for estimating the Value at Risk. No distributional assumptions are needed for the application of this method as in this case instead of modeling the whole distribution, the quantile is modelled directly. Let the θ-quantile of the distribution of portfolio returns at time t be denoted as $q_t(\boldsymbol{\beta}) \equiv (f(x_t), \boldsymbol{\beta}_\theta)$

The very general CAViaR specification is of the form:

$$q_t(\boldsymbol{\beta}) = \beta_0 + \sum_{i=1}^{p} \beta_i \, q_{t-1}(\boldsymbol{\beta}) + l\left(\beta_{p+1}, ..., \beta_{p+q}; \Omega_{t-1}\right). \tag{3}$$

Where, Ω_{t-1} represents all the available information at time t, $q_{t-1}(\boldsymbol{\beta})$ is the autoregressive term that ensures that the quantile changes smoothly over time and $l(\bullet)$ is used to connect $q_t(\boldsymbol{\beta})$ with the observable variables in the information set.

It is important to note that the process in (3) does not explode as long as the roots of $1 - \beta_1 z - \beta_2 z^2 - ... - \beta_p z^p = 0$ satisfy the condition $|z| > 1$.

A special case of CAViaR model can be defined as: $q_t(\boldsymbol{\beta}) = x^T\boldsymbol{\beta} + \varepsilon_t$, and quantile regression can be applied to estimate the vector of unknown parameters. The regression quantile is defined as the value of β that minimises the form:

$$\min_{\boldsymbol{\beta}\in\Re}\left[\sum_{t\in\left\{t:y_t\geq x_t\boldsymbol{\beta}\right\}}\theta|y_t - x_t\boldsymbol{\beta}| + \sum_{t\in\left\{t:y_t,<x_t\boldsymbol{\beta}\right\}}(1-\theta)|y_t - x_t\boldsymbol{\beta}|\right].$$

4 Bayesian Quantile Regression

The use of Bayesian inference in generalized linear and additive models is quite standard these days. Unlike conventional methods, Bayesian inference provides the entire posterior distribution of the parameters under investigation and it allows the uncertainty factor to be taken into account when making predictions. Bayesian inference is widely used nowadays, especially since, even in complex problems, the posterior distribution can be easily obtained using Markov Chain Monte Carlo (MCMC) methods. (Yu and Moyeed, 2001).

Yu and Moyeed (2001) have shown that minimisation of the check function in (2) is equivalent to the maximisation of a likelihood function formed by combining independently distributed asymmetric Laplace densities. That is, under the framework of generalized linear models, to make Bayesian inference for the conditional quantile $q_\theta(y_t|x_t)$, the following assumptions must be made: $f(y_t;\mu_t)$ is following an asymmetric Laplace distribution with probability density function $f_\theta(u) = \theta(1-\theta)\ \exp\{-\rho_\theta(u)\}$ and for some known link function g, $g(\mu_t) = \mathbf{x}^T\boldsymbol{\beta}(\theta) = q_\theta(y_t|x_t)$, for $0 < \theta < 1$.

Given the data y_t the posterior distribution of $\boldsymbol{\beta}$, $p(\boldsymbol{\beta}|\mathbf{y})$ is given by:

$$p(\boldsymbol{\beta}|\mathbf{y}) = L(\mathbf{y}|\boldsymbol{\beta})p(\boldsymbol{\beta}), \tag{4}$$

where $p(\boldsymbol{\beta})$ is the prior distribution of $\boldsymbol{\beta}$ and $L(\mathbf{y}|\boldsymbol{\beta})$ is the likelihood function defined as:

$$L\left(\mathbf{y}|\boldsymbol{\beta}\right) = \theta^n(1-\theta^n)\exp\left\{-\sum_t \rho_\theta(y_t - \mathbf{x^T}\boldsymbol{\beta})\right\}. \tag{5}$$

A standard conjugated prior is not available for quantile regression, but the posterior distribution of unknown parameters can be easily obtained using Markov Chain Monte Carlo (MCMC) methods. In theory, we could use any prior for $\boldsymbol{\beta}$, but if no realistic information is available improper uniform prior distributions for all the components of $\boldsymbol{\beta}$ are also suitable.

5 The Bayesian CAViaR Model

In this section we present our methodology for making inferences about the CAViaR model under the Bayesian Quantile regression framework.

We consider the model:

$$y_t = q_t\left(\boldsymbol{\beta}\right) + \varepsilon_t,$$

$$q_t\left(\boldsymbol{\beta}\right) = \beta_0 + \sum_{i=1}^{p}\beta_i q_{t-1}\left(\boldsymbol{\beta}\right) + l\left(\beta_{p+1},...,\beta_{p+q};\Omega_{t-1}\right). \tag{6}$$

Examples of CAViaR process have been presented by Engle and Manganelli (2004). For example, the Symmetric Absolute Value Model:

$$q_t(\boldsymbol{\beta}) = \beta_0 + \beta_1 q_{t-1}(\boldsymbol{\beta}) + \beta_2 |y_{t-1}|,$$

where, y_t denotes the vector of observations at time t and ε_t denotes the model error terms whose θ^{th} quantile is assumed to be zero, $\left(\int_{-\infty}^{0} f_\theta(\varepsilon) d\varepsilon = \theta \right)$, where $f_\theta(\bullet)$ denotes the error density.

Our aim is to demonstrate how non-parametric Bayesian Quantile regression can be used to estimate the unknown parameters in CAViaR models. These estimates will be then used to estimate the one-step ahead Value at Risk forecasts for different quantile values. Kottas and Krnjajic (2005) used a flexible nonparametric model for the prior models of the error density $f_\theta(\bullet)$by applying a nonparametric error distribution based on the Dirichlet Process (DP) mixture models (Fergguson, 1973, Antoniak, 1974). The only parametric family that has been proven suitable to use for quantile regression is the asymmetric Laplace distribution (Yu and Moyeed, 2001). Kottas and Krnjajic (2005) extended the parametric class of distribution in (5) though appropriate mixing.

A general Bayesian nonparametric setting in terms of DP mixture is given by

$$\begin{aligned}
y_t \,\big|\, \boldsymbol{\beta}, \sigma_t &\overset{ind}{\sim} Kp\big(y_t - q_t\big(\boldsymbol{\beta}\big), \sigma_t\big), t = 1..n. \\
\boldsymbol{\beta}_n &\propto 1, n = 1..p+q \\
\sigma_t \,\big|\, G &\overset{iid}{\sim} G, t = 1..n \\
G \,\big|\, M, d &\sim DP(MG_0) \\
G_0 &= IG(c, d)
\end{aligned} \tag{7}$$

where, M is the precision parameter and IG denotes an Inverse Gamma distribution with mean $\frac{d}{c-1}$.

We chose independent improper uniform priors for all the components of $\boldsymbol{\beta}$, a DP prior distributions for σ_t, $c = 2$ and d =average of the previous time series of σ_t.

The first step is to construct the joint posterior distribution for the unknown parameters which, according to the theory of Bayesian inference (4), is given by:

$$f(\boldsymbol{\beta}, \sigma_t | \mathbf{y}) \propto p(\boldsymbol{\beta}) p(\sigma_t) \prod f(y_t | \boldsymbol{\beta}, \sigma_t). \tag{8}$$

Having defined the joint posterior distribution the next step is to specify the likelihood function $f(y_t|\boldsymbol{\beta}, \sigma_t)$, define suitable prior distributions $p(\boldsymbol{\beta})$ and $p(\sigma_t)$ for the unknown parameters and then work out the full conditional posterior distribution for each of the unknown parameters.

The likelihood function $f(y_t|\boldsymbol{\beta}, \sigma_t)$ is given by:

$$f(y_t|\boldsymbol{\beta}, \sigma_t) \propto \prod f(y_t | y_{t-1}, q_{t-1}(\boldsymbol{\beta}), \boldsymbol{\beta}, \sigma_t),$$

where $f(y_t|y_{t-1}, q_{t-1}(\boldsymbol{\beta}), \boldsymbol{\beta}, \sigma_t)$ is asymmetric Laplace probability density function (Yu and Moyeed, 2001, Kottas and Krnjajic, 2005) defined as:

$$f(y_t|y_{t-1}, q_{t-1}(\boldsymbol{\beta}), \boldsymbol{\beta}, \sigma_t) = \frac{\theta(1-\theta)}{\sigma_t} \exp\left\{ -\frac{|y_t - q_t(\boldsymbol{\beta})| + (2\theta - 1)(y_t - q_t(\boldsymbol{\beta}))}{\sigma_t} \right\}. \tag{9}$$

The full conditional for $\boldsymbol{\beta}$ is obtained by isolating the terms depending on $\boldsymbol{\beta}$ from $f(\boldsymbol{\beta}, \sigma_t|\mathbf{y})$ which results in:

$$f(\boldsymbol{\beta}|others) \propto p(\boldsymbol{\beta}) \prod f(y_t | y_{t-1}, q_{t-1}(\boldsymbol{\beta}), \boldsymbol{\beta}, \sigma_t) \propto \prod Kp(y_t - (q_t(\boldsymbol{\beta})), \sigma_t).$$

6 Proper posterior

As we see from Section 5 above, a standard conjugate prior distribution is not available for the CAViaR formulation, MCMC methods may be used to draw samples from the posterior distributions. This, principally, allows us to use virtually any prior distribution. However, we should select priors that yield proper posteriors.

In this section we show that we can choose the prior $p(\boldsymbol{\beta})$ from a class of known distributions, in order to get proper posteriors.

The likelihood $f(y_t|\boldsymbol{\beta})$ in (9) is not continuous on the whole real line, but has a finite or a countably infinite set of discontinuities, thus is Riemann integrable.

First, the posterior is proper if and only if

$$0 < \int_{R^{p+q+1}} f(\boldsymbol{\beta}|\mathbf{y}) d\boldsymbol{\beta} < \infty. \tag{10}$$

Theorem 1: *Assume that the prior for $\boldsymbol{\beta}$ is improper and uniform, i.e. $p(\boldsymbol{\beta}) \propto 1$, then all posterior moments exist.*

Proof: We need to prove that

$$\int_{R^{p+q+1}} \prod_{j=0}^{p+q} |\beta_j|^{r_j} \exp\left\{ - \sum_{t=1}^{n} \frac{|y_t - q_t(\boldsymbol{\beta})| + (2\theta - 1)(y_t - q_t(\boldsymbol{\beta}))}{\sigma_t} \right\} d\boldsymbol{\beta} \tag{11}$$

is finite, where $(r_0, ..., r_j)$ denote the order of the moments of $\boldsymbol{\beta} = (\beta_0, ..., \beta_p)$ and $\mathrm{q}_t(\boldsymbol{\beta})$ is the general CAViaR (3), which can be re-represented as

$$q_t(\boldsymbol{\beta}) = \beta_0 \{1 + \sum_t \prod_k \beta_1^{k_1} \beta_2^{k_2} ... \beta_{p+q}^{k_{p+q}}\} + l(\beta_{p+1}, ..., \beta_{p+q}; \Omega_{t-1})$$

where k_i $(i = 1, ..., p+q)$ are some no-negative integers.

By making the integral transformation $\alpha = \beta_0 \{1 + \sum \prod \beta_1^{k_1} \beta_2^{k_2} ... \beta_{p+q}^{k_{p+q}}\}$, $\beta_i = \beta_i$ for $i = 1, .., p+q$, we obtain $q_t(\boldsymbol{\beta}) = \alpha + l(\beta_1, ..., \beta_{p+q}; \Omega_{t-1})$

Note that

$$\sum_{t=1}^{n} \frac{|y_t - q_t(\boldsymbol{\beta})| + (2\theta - 1)(y_t - q_t(\boldsymbol{\beta}))}{\sigma_t} = c_1 \sum_{t=1}^{n} |y_t - q_t(\boldsymbol{\beta})| + c_2 \sum_{t=1}^{n} (y_t - q_t(\boldsymbol{\beta}))$$

$$= c_1 \sum_{t \in \ell} (y_t - q_t(\boldsymbol{\beta})) - c_1 \sum_{t \notin \ell} (y_t - q_t(\boldsymbol{\beta})) + c_2 \sum_{t=1}^{n} (y_t - q_t(\boldsymbol{\beta}))$$

where c_1 and $c_2 > 0$ and the set $\ell = \{t : q_t(\boldsymbol{\beta}) > 0\}$.

It is sufficient to prove that $\int_{R^{p+q+1}} \prod_{j=0}^{p+q} |\beta_j|^{r_j} \exp\left\{ - \sum_{t \in \ell} (y_t - q_t(\boldsymbol{\beta})) \right\} d\boldsymbol{\beta}$ is finite. According to Lemmas 1 of Yu and Stander (2007) this is true if and only if

$\int_{R^{p+q+1}} \prod_{j=0}^{p+q} |\beta_j|^{r_j} g(h(\theta) \sum_{t \in \ell} (y_t - q_t(\boldsymbol{\beta})) d\boldsymbol{\beta}$ is finite, where $h(\theta) = \theta(1-\theta)/\sigma_t$ and $g(T) = \exp(-|T|)$, which is true according to Lemma 2 of Yu and Stander (2007).

7 Simulations

7.0.1 Symmetric Absolute Value Model

To check whether our proposed methodology is able to produce consistent estimates we have run several Monte Carlo simulations.

Take Symmetric Absolute Value Specification as an example:

$$y_t = \beta_0 + \beta_1 q_{t-1}(\boldsymbol{\beta}) + \beta_2 |y_{t-1}| + \varepsilon_t.$$

From $q_t(\boldsymbol{\beta}) = \beta_0 + \beta_1 q_{t-1}(\boldsymbol{\beta}) + \beta_2 |y_{t-1}|$ and $q_1 = \beta_0$ this model can be reformulated as

$$y_t = B_0 + B_1 |y_{t-2}| + B_2 |y_{t-1}| + \varepsilon_t$$

where $B_0 = 1 + \beta_1 + \beta_1^2$, $B_1 = \beta_1\beta_2$, and $B_2 = \beta_2$

We have used the latter model to test our methodology. We considered the model:

$$y_t = 1 + 0.05 q_{t-1} + 0.6 |y_{t-1}| + \varepsilon_t,$$

and assumed $\epsilon_t \sim N(0,1)$, for all $t = 1, ..., 600$.

By reformulating our model we obtained: $y_t = 1 + 0.03 |y_{t-2}| + 0.6 |y_{t-1}| + \varepsilon_t$

We have generated 600 observations from this model and we estimated the parameters using the Symmetric Absolute Value process as quantile specification. We estimated the parameters for different quantile values, namely, 1% 5% 25%, 75% 95% and 99% . We run the MCMC algorithm for 150,000 iterations to make sure their convergence and mixing then discarded the first 100,000. The value recorded for each parameter was the mean of the values obtained in the last 50,000 iterations.

The results are shown in Table 1.

Table 1: Obtained Results for Symmetric Absolute Value

θ	$\mathbf{B}_0$	$\hat{B}_0$	r1	$\mathbf{B}_1$	$\hat{B}_1$	r2	$\mathbf{B}_2$	$\hat{B}_2$	r3
0.01	-1.23	-1.17(sd.0.1)	0.28	0.03	-0.08 (sd.0.04)	0.21	0.6	0.74 (sd.0.03)	0.21
0.05	-0.64	-0.53(sd.0.1)	0.22	0.03	-0.04 (sd.0.05)	0.18	0.6	0.67 (sd.0.05)	0.17
0.25	0.33	0.23 (sd.0.1)	0.23	0.03	0.08(sd.0.04)	0.16	0.6	0.59 (sd.0.05)	0.16
0.75	1.67	1.53 (sd.0.1)	0.25	0.03	0.08 (sd.0.04)	0.20	0.6	0.62 (sd.0.04)	0.18
0.95	2.64	2.71 (sd.0.1)	0.22	0.03	0.04 (sd.0.05)	0.18	0.6	0.58 (sd.0.1)	0.18
0.99	3.33	2.74 (sd.0.1)	0.27	0.03	0.28 (sd.0.03)	0.20	0.6	0.58 (sd.0.04)	0.17

As expected the worse results were obtained for the extreme quantile values, 1% and 99% , since in a sample of 600 observations, it is very difficult to get precise estimates.

The results of the simulations for the other quantile values were pretty close to the real values. The plots of the posterior distributions of the estimated parameters showed dominant modes close to the real values of θ. The quality of the estimates was checked using the acceptance rate (r1, r2 and r3 in Table 1), which for all the parameters were in the acceptable range and close to the optimal acceptance rate (Roberts and Rosenthal, 2001).

8 Applications

8.0.2 Comparison between Classical Quantile Regression (CQR) and Bayesian Quantile Regression (BQR

In order to make comparisons between Classical Quantile Regression (CQR) and Bayesian Quantile Regression (BQR) we carried out analysis on real data series using both methods. Our sample consisted of monthly prices for the NASDAQ Composite Index for the period from April 1971 to December 1998. Daily returns were computed as 10 times the difference of the logs of the prices. The parameters were estimated for different quantile values using the Symmetric Absolute Value CAViaR specification. The results are shown in tables 2 and 3. As it can be seen from the results the estimates of the parameters obtained by BQR are very similar to the results obtained using CQR for most of the quantile values.

Table 2: **Obtained Results, CQR**

VaR	β_0	β_1	β_2
5%	-0.04	1.4	-0.4
10%	-0.04	0.8	-0.4
25%	0.0	7	-0.03
50%	0.0	7	-0.01
75%	0.08	0.4	0.2
90%	0.17	0.2	0.3
95%	0.23	0.2	0.3

Table 3: **Obtained Results, BQR**

VaR	β_0	β_1	β_2
5%	-0.09	1.6	-0.5
10%	-0.1	1	-0.4
25%	0.03	1	-0.2
50%	0.0	6	-0.01
75%	0.08	0.4	0.2
90%	0.15	0.3	0.2
95%	0.14	0.7	0.3

9 Summary and Future Work

The aim of this paper was to demonstrate a new alternative approach for estimating the VaR for portfolio returns. Engle and Manganelli (2004) proposed a semi-parametric approach to VaR calculation, the Conditional Autoregressive Value at Risk (CAViaR) model, which is a very popular method of estimation in which the quantile is modelled directly and no distributional assumptions are necessary. We have demonstrated how non-parametric Bayesian quantile regression can be used for the estimation of VaR under the CAViaR framework. We demonstrated our approach using a simulated example for the Symmetric Absolute Value model. Furthermore we proceeded to a comparison of our approach with classical CAViaR. The results of both the simulations and the comparison were promising therefore our future work in this area will focus on the application of our methodology for estimation of VaR in real data series, including exploration of prior selection and model comparison.

References

[1] Antoniak, C. E. (1974). *Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems*. The Annals of Statistics, vol. 2, pp. 1152-1174.

[2]Chen, M. and Chen, J.(2003). *Application of Quantile Regression to estimation of value at Risk*. Working Paper.

[3] Engle, R.F. & S. Manganelli (2004). *CAViaR: Conditional autoregressive Value at Risk by regression quantile*. Journal of Business and Economic Statistics, 22, 367-381.

[4]Ferguson, T. S. (1973). *A Bayesian analysis of some nonparametric problems*. The Annals of Statistics, vol. 1, pp.209-230.

[5]Giot, P. & Laurent, S. (2004). *Modelling daily Value-at-Risk using realized volatility and ARCH type models*. Journal of Empirical Finance, Elsevier, vol. 11(3), pp. 379-398.

[6]Hull, J.C. (2000), *Options, Futures, and Other Derivatives* (Fourth Edition), Prentice Hall.

[7]Koenker, R & Hallock, K.F (2001). *Quantile Regression* . Journal of Economic Perspective. vol. 15(4), pp. 143-156.

[8]Kottas, A., & Krnjajic, M. (2005). *Bayesian Nonparametric Modeling in Quantile Regression*. Technical Report.

[9] Roberts G. O.& Rosenthal J.S. (2001). *Optimal Scalling for Various Metropolis-Hasting Algorithms*. Statistical Science. vol. 16(4), pp. 351-367

[10]Manganelli, S. & Engle, R. (2004). *A Comparison of Value at Risk Models in Finance*. Risk Measures for the 21st Century, ed. Giorgio Szego, Wiley Finance.

[11] Yu, K. & Stander, J. (2007). *Bayesian Analysis of a Tobit Quantile Regression model*. Journal of Econometrics, vol. 137, pp.260-276.

[12] Yu, K., Lu, Z. & Stander, J. (2003). *Quantile Regression: Applications and Current Research Areas*. The Statistician, vol. 52(3), pp.331-350.

[13] Yu, K. & Moyeed, R.A. (2001). *Bayesian Quantile Regression*. Statistics and Probability Letters, vol. 54, pp. 437-447

Is that really the pattern we're looking for? Bridging the gap between statistical uncertainty and dynamic programming algorithms in pattern detection

John A. D. Aston
CRiSM, Dept of Statistics
University of Warwick, UK
and
Institute of Statistical Science
Academia Sinica, Taiwan
j.a.d.aston@warwick.ac.uk

Michael Jyh-Ying Peng
Computer Science and Information Engineering
National Taiwan University, Taiwan
and
Institute of Statistical Science
Academia Sinica, Taiwan
jypeng@stat.sinica.edu.tw

Donald E. K. Martin
Department of Statistics
North Carolina State University, USA
martin@stat.ncsu.edu

Abstract

Two approaches to statistical pattern detection, when using hidden or latent variable models, are to use either dynamic programming algorithms or Monte Carlo simulations. The first produces the most likely underlying sequence from which patterns can be detected but gives no quantification of the error, while the second allows quantification of the error but is only approximate due to sampling error. This paper describes a method to determine the statistical distributions of patterns in the underlying sequence without sampling error in an efficient manner. This approach allows the incorporation of restrictions about the kinds of patterns that are of interest directly into the inference framework, and thus facilitates a true consideration of the uncertainty in pattern detection.

1 Introduction

Dynamic programming algorithms such as the Viterbi algorithm (Viterbi 1967) provide the mainstay of much of the literature on pattern recognition and classification, especially when dealing with Hidden Markov Models (HMMs) and other related models. Patterns often consist of functions of unobserved states and as such as not predicted directly by the model, but indirectly through analysis of the underlying states themselves. In Viterbi analysis, a trained model is used to analyse test data, and the most probable underlying sequence, the Viterbi sequence, is determined and then treated as deterministically correct. This Viterbi sequence is then used to search for patterns of interest that might or might not have occurred in the data. If the patterns occur in the Viterbi sequence, they are deemed present in the data, otherwise not. However, there are usually restrictions on the types of patterns that can occur, and when these restrictions are not met, possible patterns in the underlying sequence are either discarded or altered to make them fit the known restrictions. However, this inherently alters the nature of the sequence (as discarding or altering states alters the complete underlying sequence), rendering it not only different from that predicted from the dynamic programming algorithm but also destroying the feature of the underlying sequence being most probable, even amongst those sequences that satisfy the restrictions.

An alternative approach to the problem of pattern detection is to dispense with the dynamic programming algorithm and instead use approximate methods based on statistical sampling of the underly-

ing sequence. Monte Carlo samples of the underlying sequence can be drawn, often from efficient Markov chain algorithms (Cappé, Moulines, and Rydén 2005), and then functions of these states used to make inferences about the presence of patterns or not. However, approximation algorithms have the inherent disadvantage of being by nature approximate. It is also often difficult to determine the number of samples needed to make accurate classifications for functions of the underlying states, especially for data with a very large number of observations.

This paper describes a method which addresses the disadvantages of the methods above. It generates statistical distributions associated with the patterns and the model which are exact (in that they are not subject to sampling error). However, the method also allows the patterns to be explored with reference to all the possible combinations of underlying sequence and can be easily extended to discount any sequences that do not possess the required known restrictions. They are inherently fast and efficient, with computational complexity only growing linearly with the size of the restriction and the number of possible patterns present.

The paper continues as follows. Firstly a brief section outlining some notation is given. Then in Section 3, a well known example from bioinfomatics relating to the analysis of CpG islands in DNA nucleotide sequences is given as motivation for the techniques. The theory underlying the procedure is then given in Section 4. Section 5 returns to the CpG island example to examine the gains of using the new methodology, while the last section gives some discussion and ideas for extensions.

2 Notation

The methods which will be examined here can be applied to Markov switching models with the general form:

$$\begin{aligned} y_t &\sim f(S_{t-r:t}, y_{1:t-1}), \\ P[S_t|S_{-r+1:t-1}] &= P[S_t|S_{t-1}], \quad t = 1, \ldots, n, \quad S_t \in \mathcal{S} \end{aligned} \tag{1}$$

The data, y_t, from time 1 to time n, which can be either discrete or continuous, is distributed conditional on previous data and r previous switching states $S_{t-r}, \ldots, S_{t-1}$ in addition to the current state S_t (as well as other parameters required for the model, the values of which are implicitly assumed to be fixed). The common definition of a HMM is the special case with y_t dependent on S_t only (in this case r is set to 1 due to the underlying Markov chain rather than the data dependence). The notation $y_{t_1:t_2} = y_{t_1}, \ldots, y_{t_2}$ is introduced and used from here on, with $S_{t_1:t_2}$ defined analogously. This general form is equivalent to assumption **Y2** in Frühwirth-Schnatter (2006, p. 317). For simplicity, the switching states $\{S_t\}$ are assumed to be a first-order Markov chain with finite state space $\mathcal{S}$, but extension to higher-order Markov structures is straightforward. A given initial distribution for $S_{-r+1:0}$ is also assumed. With suitable modification, the above model may also include exogenous variables. No assumption on the distribution of the noise in the system is made other than that the smoothed probabilities of the states conditional on the data must exist.

3 Motivation - CpG Island Analysis

The use of HMMs to model DNA sequences with heterogenous segments was pioneered by Churchill (1989). HMMs have been shown to be especially suitable for the analysis of CpG islands. A CpG island is a short segment of DNA in which the frequency of CG pairs is higher than in other regions. The "p" indicates that C and G are connected by a phosphodiester bond. The C in a CG pair is often modified by methylation, and if that happens, there is a relatively high chance that it will mutate to a T, and thus CG pairs are under-represented in DNA sequences. Upstream from a gene, the methylation process is suppressed in a short region of length 100-5,000 nucleotides, known as CpG islands (Bird 1987). The underlying nucleotide generating sequence can be modelled as two different systems, one for when the sequence is in a CpG island, and one for when it is not.

As CpG islands can be especially useful for identifying genes in human DNA (Takai and Jones 2002), different methods have been developed for their detection. One method of determining islands is to use HMMs for the analysis (Durbin et al. 1998). Software is readily available to implement these HMM-based methods for CpG island analysis, for example Guéguen (2005). The Viterbi algorithm is used to segment the sequence and analysis then proceeds as indicated above using the "deterministic" Viterbi sequence.

Define $\mathcal{S} = \{A^+, C^+, G^+, T^+, A^-, C^-, G^-, T^-\}$, where a superscript "+" indicates that the state is within a CpG island and "−" that it is not, and let $\mathcal{S}_Y = \{A, C, G, T\}$ be the state space of the data, which in this case is discrete. The transition probability matrix associated with the state sequence $\{S_t\}$ is taken to be that in Aston and Martin (2007) which was based on the transition probability matrices given in Durbin et al. (1998). These were calculated using maximum likelihood methods from human DNA with 48 putative CpG islands present (which were predetermined using other methods).

If Q represents a generic nucleotide, i.e. $Q \in \{A, C, G, T\}$, then $P(y_t = Q|S_t = Q^-) = P(y_t = Q|S_t = Q^+) = 1$. Even though the observed nucleotide is totally determined if the underlying state is known, it is not possible to know whether an observation came from a CpG island or not. Finally, define the initial distribution as $\pi(A^-) = \pi(C^-) = \pi(G^-) = \pi(T^-) = \frac{1}{4}$, i.e. the underlying state sequence is equally likely to start in any of the non-CpG island states.

Take as an example the sequence from Human DNA, chromosome 20, locus AL133339 (Barlow 2005) with 18,058 base pairs (bps). Using the method based on the Viterbi algorithm (Durbin et al. 1998) four islands were identified, however, only two of them are at least 100 bps in length, a requirement of the biological definition of a CpG island. This immediately raises a question. If only sequences with islands of length at least 100 should be identified, how should this underlying sequence be altered to account for this fact? Deleting any islands of length less than 100 may lead to sequences that are less probable than extending those same islands so that their length is longer than 100. This suggests that the condition on the islands being of length at least 100 should be an integral part of the analysis rather than a postprocessing step.

4 Analysis of Runs and Patterns in HMMs and related models

A *simple pattern* Λ_i refers to a specified sequence of symbols of $\mathcal{S}$, where the symbols are allowed to be repeated. A *compound pattern* Λ is the union of simple patterns, i.e. $\Lambda = \cup_{i=1}^{\eta} \Lambda_i$, where the lengths of the simple patterns Λ_i may vary, $\Lambda_a \cup \Lambda_b$ denotes the occurrence of pattern Λ_a or pattern Λ_b, and the integer $\eta \geq 1$.

Consider now a system $\Lambda^{(1)}, \ldots, \Lambda^{(c)}$ of c compound patterns, $c \geq 1$, with corresponding numbers $r_1, \ldots, r_c$, where r_j denotes the required number of occurrences of compound pattern $\Lambda^{(j)}$. If the waiting time of interest is the time until the first occurrence of one of the compound patterns its specified number of times, $\Lambda^{(1)}, \ldots, \Lambda^{(c)}$ are called *competing patterns* (Aston and Martin 2005). If all of the patterns must occur their specified number of times, the system is called *generalised later patterns* (Martin and Aston 2008).

A run of length k in state s is defined to be the consecutive occurrence of k states that are all equal to s, i.e. $S_{t-k+1} = s, \ldots, S_t = s$ for some t. Of particular interest in this paper will be this special type of pattern and from here on, for simplicity, only runs will be considered, although all the results are equally valid for the other types of patterns above.

4.1 Waiting time distributions for runs

Define $W_s(k, m)$ to be the waiting time of the mth run of length at least k in state s and let $W(k, m)$ be the waiting time for the mth run of length at least k of any state, where all the runs are not required to be of the same state.

To determine distributions associated with runs, finite Markov chain imbedding methodology (Fu and Koutras 1994) will be used. The idea involves imbedding the $\{S_t\}$ process into a new Markov process Z_t with a larger state space.

The state space of Z_t will consist of vector states of the form $((s_1, \ldots, s_r), j)$, $s_1, \ldots, s_r \in \mathcal{S}$, $j = 0, \ldots, k$, consisting of an r-tuple giving the current and previous $r - 1$ states of the switching process (i.e. $S_t, \ldots, S_{t-r+1}$), and a component j that counts the current observed run length.

The state space will change slightly depending on whether it is being used to calculate $P[W_s(k,1) \leq t]$ or $P[W(k,1) \leq t]$. When calculating $P[W_s(k,1) \leq t]$, for $k > r$, the state space $\mathcal{Z}_s$, for Z_t, is

$$\mathcal{Z}_s = \left(\bigcup_{s_r \in \mathcal{S}} \cdots \bigcup_{s_1 \in \mathcal{S}} ((s_1, s_2, \ldots, s_r), 0) \right) \cup \left(\bigcup_{s_r \in \mathcal{S}} \cdots \bigcup_{s_2 \in \mathcal{S}} ((s, s_2, \ldots, s_r), 1) \right)$$
$$\cup \ldots \cup \left(\bigcup_{s_r \in \mathcal{S}} ((s, \ldots, s, s_r), r-1) \right) \cup \left(\bigcup_{i=r}^{k-1} ((s, \ldots, s), i) \right) \cup ((s, \ldots, s), k), \tag{2}$$

while it is defined similarly for $k \leq r$. The state $((s, \ldots, s), k)$ in (2) corresponds to an absorbing state, and indicates that a run of length at least k has occurred. The notation A will be used to denote the class of absorbing states (when $k < r$ multiple absorbing states could be present). The state space $\mathcal{Z}$ for calculating $P[W(k,1) \leq t]$ is then just

$$\mathcal{Z} = \bigcup_{s \in \mathcal{S}} \mathcal{Z}_s. \tag{3}$$

The $\{Z_t\}$ chain conditioned on the data is inhomogeneous, as $P[S_t | S_{t-r:t-1}, y_{1:n}]$ is time-varying. Specifically the transition probabilities between Z_t states are governed purely by the smoothed transition probabilities for the first component of the Z_t vector states, which is itself an r-tuple of S_t states.

For transient states in $\mathcal{Z}_s$ or $\mathcal{Z}$, there are only $|\mathcal{S}|$ possible transitions. The transition probabilities for a generic state $((s_1, \ldots, s_r), i)$ to a new state $((s_0, s_1, \ldots, s_{r-1}), j)$ can be determined in the following steps. The non-zero generic transition probability in the $z^* \times z^*$ transition probability matrix M_t (where z^* denotes $|\mathcal{Z}_s|$ or $|\mathcal{Z}|$, as appropriate) for the Z_t process is

$$\begin{aligned} P[Z_t = ((s_0, s_1, \ldots, s_{r-1}), j) | Z_{t-1} = ((s_1, \ldots, s_r), i), y_{1:n}] \\ = P[S_t = s_0 | S_{t-1} = s_1, \ldots, S_{t-r} = s_r, y_{1:n}] \end{aligned} \tag{4}$$

for particular values of i and j that are consistent with the possible state transitions, otherwise they are zero. See Aston and Martin (2007) for a related construction of transition probabilities which can be extended to this case.

The initial probability distribution $\psi_0 = P[Z_0]$ is contained in a $1 \times z^*$ row vector. The non-zero probabilities can be set to the initial distribution for the S_t process

$$P[Z_0 = ((S_0, S_{-1}, \ldots, S_{-r+1}), 0)] = P[S_0, S_{-1}, \ldots, S_{-r+1}]. \tag{5}$$

With the state space for Z_t constructed in this way, the $1 \times z^*$ probability vector of being in any state of Z_t at time t, ψ_t, is given by

$$\psi_t = \psi_0 \prod_{j=1}^{t} M_j, \tag{6}$$

which follows from the well-known Chapman-Kolmogorov equations for Markov chains. The distributions of interest $P[W_s(k,1) \leq t]$ or $P[W(k,1) \leq t]$ can then be calculated as

$$P[W_s(k,1) \leq t] = P[Z_t \in A] = \psi_t U(A), \tag{7}$$

with the analogous result holding for $P[W(k,1) \leq t]$, where $U(\Omega)$ is a $z^* \times 1$ column vector with ones in the locations of the members of the set Ω and zeros otherwise.

A class of states C, called continuation states, are added to $\mathcal{Z}_s$ and $\mathcal{Z}$, and the definition of z^* is updated to include the continuation states. The role of the continuation states is that once the ith run of length at least k has occurred, $i = 1, \ldots, m-1$, (where it is necessary for m runs to occur), a new Markov chain $\{Z_t^{(i+1)}\}$ is started to determine the probabilities associated with the next occurrence of a run of the desired length. The continuation states serve to initialise the new chain $\{Z_t^{(i+1)}\}$, and indicate that run i is still in progress and that the run needs to end before the $(i+1)$st run can possibly begin.

The continuation states correspond in an one-to-one fashion with the absorbing states. The (less than full rank) $z^* \times z^*$ matrix Υ, defined on the redefined state space $\mathcal{Z}_s$ or $\mathcal{Z}$ which have been augmented with the continuation states, with elements

$$\Upsilon(z_1, z_2) = \begin{cases} 1 & \text{if } z_1 \in A \text{ and } z_2 \in C \text{ is the corresponding state} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

is used to map probabilities of being in the states of A into probabilities for being in the corresponding states in C.

The transition probability matrices M_t are revised to account for the continuation states. Continuation states may only be entered from other continuation states. The general non-zero transition probabilities beginning in a continuation state $((s_1, \ldots, s_r), -1) \in C$ where the -1 indicates the previous run is still in progress, and conditional on the data are of the form

$$\begin{aligned} P[Z_t = ((s_0, s_1, \ldots, s_{r-1}), j) | Z_{t-1} = ((s_1, \ldots, s_r), -1), y_{1:n}] \\ = P[S_t = s_0 | S_{t-1} = s_1, \ldots, S_{t-r} = s_r, y_{1:n}], \end{aligned} \tag{9}$$

for appropriate values of $j \in \{-1, 0, 1\}$ depending on the value of s_0. The transition probabilities for the other states in either $\mathcal{Z}_s$ or $\mathcal{Z}$ are unchanged.

Let Ψ_0 be a $m \times z^*$ initial matrix consisting precisely of ψ_0 stacked upon $m-1$ row vectors of zeros, and define the $m \times z^*$ matrix $\Psi_t = [\psi_t^{(1)'} \ldots \psi_t^{(m)'}]'$ where $'$ denotes transpose of the $1 \times z^*$ row vectors $\psi_t^{(i)}$. The Markov transition, used in (6), is updated to an algorithm, which at each time t, $t = 1, \ldots, n$, has the following two steps:

$$\Psi_t = \Psi_{t-1} M_t, \tag{10}$$

$$\psi_t^{(i)} \leftarrow \psi_t^{(i)} + \psi_{t-1}^{(i-1)} (M_t - I)\Upsilon, \quad i = 2, \ldots m, \tag{11}$$

where $\psi_t^{(i)}$ is the ith row of Ψ_t and the second term in (11) increments each row with the probability that a run has occurred at that time point. For more information on the derivation of this algorithm in the case of the common definition of HMMs and finite output states see Aston and Martin (2007).

The marginal distributions can then be found easily as

$$P[W(k,i) = t_i + k - 1] = (\psi^{(i)}_{t_i+k-1} - \psi^{(i)}_{t_i+k-2}) U(A), \tag{12}$$

and similarly for $P[W_s(k,i) = t_i + k - 1]$.

In addition, using the continuation state C, it is possible to determine the distribution of when the system leaves a run $P[W_s^e(k,i) = t]$, $i = 1, \ldots, m-1$. The waiting time $W_s^e(k,i)$ is defined to be the time that the ith run in state s ends, with $W^e(k,i)$ being analogously defined for the ith run in any state. A run is in progress while Z_t is in the continuation states, and as such when the chain leaves the states, the run is over. Thus the waiting time distribution for the end of a run to occur at time t is given by

$$P[W_s^e(k,i) = t] = \psi_t^{(i)} (I - M_{t+1}) U(C), \quad i = 1, \ldots, m-1. \tag{13}$$

This allows for a complete set of distributions to be given for the start and finish of any particular run. Again the analogous result for $P[W^e(k,i) = t]$ also holds.

Using the chains given above, it is also possible to determine the distribution of the number of runs $P[N_s(k) = i]$ into a particular state, or the number of runs $P[N(k) = i]$ which occurred in the data. This distribution is given by

$$P[N_s(k) = i] = P[W_s(k,i) \leq n] - P[W_s(k,i+1) \leq n], \quad i = 0, \ldots, \lfloor n/(k+1) \rfloor, \tag{14}$$

where $\lfloor n/(k+1) \rfloor$ indicates the integer part of $n/(k+1)$. In practice, the value at which $P[W_s(k,i) \leq n]$ becomes negligible will be $i \ll \lfloor n/k \rfloor$. This is similarly true for $P[N(k) = i]$, by considering $P[W(k,i) \leq n]$ for $i = 0, \ldots, \lfloor n/k \rfloor$.

4.2 Computational Considerations

Given the prevalence of Bayesian techniques in the analysis of Markov switching models (see Frühwirth-Schnatter (2006) for the latest on these techniques), it is of interest to compare the computational cost of calculating the waiting time distributions through the exact scheme above versus

drawing samples from the conditional distribution of states given observations. Of course in terms of error, for fixed parameters, the two approaches cannot be compared as the exact distribution is not subject to any sampling error.

For both methodologies, drawing conditional samples of the underlying states and the techniques above, a pass through a Markov chain is necessary. Every state in either approach has at most $|\mathcal{S}|$ possible transition destinations, so all that needs to be compared is the size of the state spaces associated with the two techniques.

For drawing conditional samples, a state space of size $|\mathcal{S}|^r$ is needed. For the presented computations given above, if $k > r$ then the state space $\mathcal{Z}_s$ is of size

$$\sum_{i=0}^{r} |\mathcal{S}|^i + (k - r + 1) = \frac{1 - |\mathcal{S}|^{r+1}}{1 - |\mathcal{S}|} + (k - r + 1) < |\mathcal{S}|^r |\mathcal{S}| + (k - r + 1)$$

(the number of states given in (2) plus one for the continuation state) while for $\mathcal{Z}$, the size needed is at most

$$|\mathcal{S}|^r + \frac{1 - |\mathcal{S}|^{r+1}}{1 - |\mathcal{S}|} + |\mathcal{S}|(k - r + 1) < |\mathcal{S}|^r(|\mathcal{S}| + 1) + |\mathcal{S}|(k - r + 1).$$

Thus when $k > r$, if $k \ll |\mathcal{S}|^r$, at most $|\mathcal{S}|m$ (and often less) equivalent sample computations are needed to calculate the marginal waiting time distributions for all m runs. Of course as k increases, the number of states will increase, but this is only at a linear rate proportional to k.

For $k < r$, the size of state space $\mathcal{Z}_s$ needed is

$$(\sum_{i=r-k}^{r} |\mathcal{S}|^i) + |\mathcal{S}|^{r-k}$$

while for $\mathcal{Z}$

$$|\mathcal{S}|^r + (\sum_{i=r-k+1}^{r} |\mathcal{S}|^i) + |\mathcal{S}|^{r-k+1}.$$

All these calculations presume that the state space of the model is of a general finite structure. Models such as the change point model of Chib (1998) would require significantly less computation for the exact distributional method given the structure in the model.

5 CpG Islands - revisited

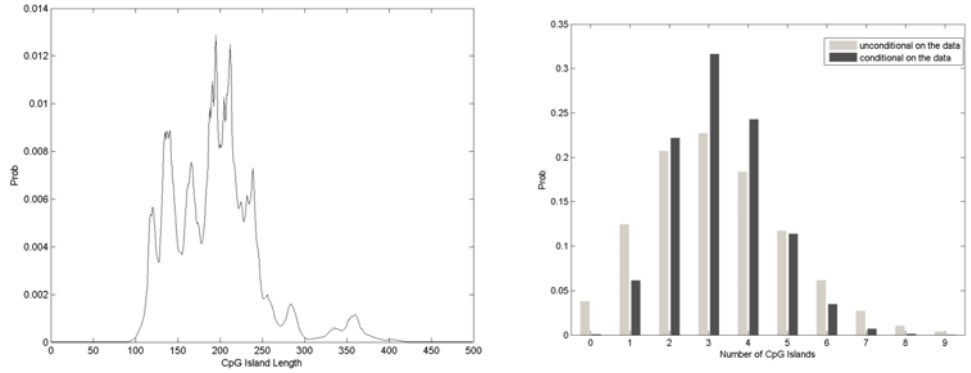
Returning to the example sequence AL133339, CpG islands can be seen as sequences of $+$'s (since + denotes $A^+ \cup C^+ \cup G^+ \cup T^+$). The maximal length and the number of islands in total in the sequence is of interest, in addition to the locations of islands of length at least 100. The definition of runs given above can easily be extended to this case.

The state space for the auxiliary $\{Z_t\}$ chain is given by

$$\bigcup_{Q \in \{A,C,G,T\}} \left\{(Q^+, 1), (Q^+, 2), \ldots, (Q^+, k-1), Q^-, (Q^+, k)\right\}, \tag{15}$$

where k is a specified run length, (Q^+, i) gives the value of S_t and the current length of the run of +'s, and (Q^+, k) are absorbing states to indicate that k consecutive +'s have occurred. The desired run occurs by time t if and only if $Z_t \in (Q^+, k)$, and thus theorems from Section 4 may be used to compute probabilities. The chain is initialised in the Q^- states.

As can be seen in Figure 1, the maximal length of CpG islands is most likely to be in the region of 200 bps for this data set under the settings of the model. However, the Viterbi sequence contains a CpG island of 362 bps, significantly longer than the maximal number likely present. This is because the Viterbi sequence only considers the whole underlying sequence rather than any function of states that leads to particular patterns, and thus does not necessarily lead to the most likely patterns. As can also be seen in Figure 1b, computing probabilities for states unconditional on the data to predict the number of CpG islands of length at least 100 bps present, so that the biological restrictions



(a) PMF of maximal CpG island length. (b) PMF of number of CpG islands of length ≥ 100.

Figure 1: CpG island distributions for the 18,058 nucleotide gene sequence locus AL133339 from Human Chromosome 20. The first plot shows the probability mass function (PMF) of the maximal CpG island length using the entire data sequence. The second plot depicts the PMF of the number of CpG islands of length at least 100 using the entire observed data sequence, as well as unconditional on the data.

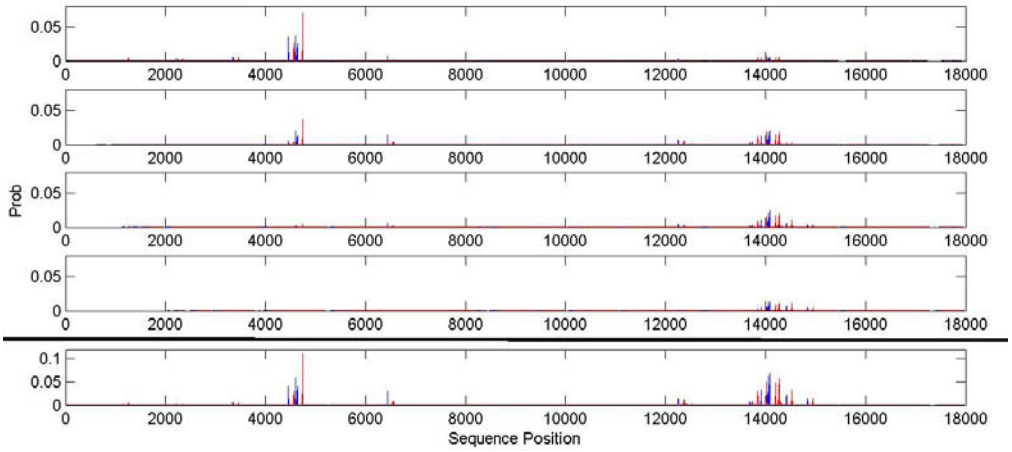


Figure 2: CpG island distributions for the 18,058 nucleotide gene sequence locus AL133339 from Human Chromosome 20. The first plot shows the probability mass function (PMF) of starting (blue) and finishing (red) position of the first CpG island, while the second graph gives the position of the second etc., until the position of the fourth. The final plot (the sum of the plots above) graphs the probability at each point of either the start or finish of any CpG island at that position.

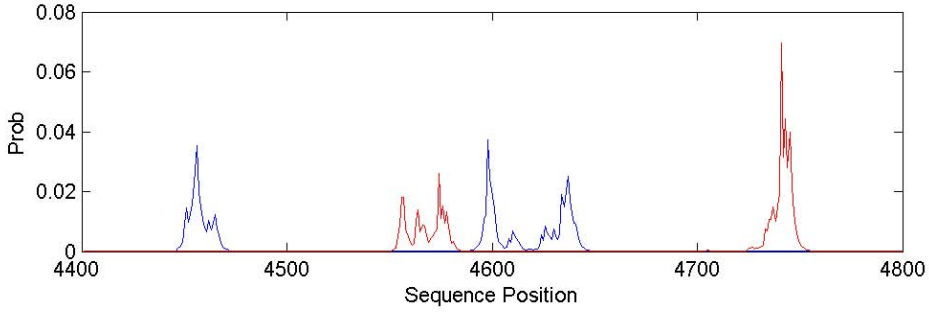


Figure 3: CpG island distributions for the 18,058 nucleotide gene sequence locus AL133339 from Human Chromosome 20. Zoom of the distribution of the position of the first CpG island around the sequence positions 4400-4800.

are satisfied, gives a similar distribution to that conditional on the actual data. However it does underestimate the likely number of CpG islands present.

Figures 2 and 3 give examples of the actual distributions of the patterns (runs of length at least 100 bps) obtained from the algorithm. Figure 2 shows the location distributions of particular occurrences of CpG islands, while the final graph shows the pointwise probabilities of CpG islands starting or finishing at various locations. While the distributions in Figure 2 seem to contain atoms at certain points and zeros otherwise, by considering particular sequence positions as in Figure 3, we see that the distributions are smoother than might be expected, yielding some uncertainty in the exact position of the CpG islands.

6 Discussion

This paper has presented a methodology to examine patterns that occur in the underlying state sequences of HMMs and related models. This differs from previous work in that the patterns themselves are integral to the methodology rather than being determined in a post processing step from either a most probable sequence or from a sampled distribution of underlying states. The methodology allows for the quantification of uncertainty about the number, maximal length, and position of patterns within a data set. This allows a statistical consideration to be made as to whether possible patterns in the data really are there or not.

The methodology here was examined with explicit reference to HMMs and Markov switching models, but can be extended to many other types of graphical models and other machine learning and theoretical computer science techniques. In addition, it would be of interest to extend the methods to stochastic patterns which means that the patterns themselves are subject to noise or a probability distribution on their exact form.

Acknowledgements

The first author gratefully acknowledges the support of EPSRC and HEFCE through the CRiSM program grant.

References

Aston, J. A. D. and D. E. K. Martin (2005). Waiting time distributions of competing patterns in higher-order Markovian sequences. *Journal of Applied Probability 42*, 977–988.

Aston, J. A. D. and D. E. K. Martin (2007). Distributions associated with general runs and patterns in hidden Markov models. *Annals of Applied Statistics 1*, 585–611.

Barlow, K. (2005). AL133339 locus of chromosome 20. EMBL/GenBank/DDBJ databases.

Bird, A. (1987). CpG islands as gene markers in the vertebrate nucleus. *Trends in Genetics 3*, 342–347.

Cappé, O., E. Moulines, and T. Rydén (2005). *Inference in Hidden Markov Models*. Springer.

Chib, S. (1998). Estimation and comparison of multiple change-point models. *Journal of Econometrics 86*, 221–241.

Churchill, G. A. (1989). Stochastic models for heterogeneous DNA sequences. *Bulletin of Mathematical Biology 51*, 79–94.

Durbin, R., S. Eddy, A. Krogh, and G. Mitchison (1998). *Biological Sequence Analysis*. Cambridge University Press.

Frühwirth-Schnatter, S. (2006). *Finite Mixture and Markov Switching Models*. Springer.

Fu, J. C. and M. V. Koutras (1994). Distribution theory of runs: A Markov chain approach. *Journal of the American Statistical Association 89*, 1050–1058.

Guéguen, L. (2005). Sarment: Python modules for HMM analysis and partitioning of sequences. *Bioinformatics 21*, 3427–3428.

Martin, D. E. K. and J. A. D. Aston (2008). Waiting time distribution of generalized later patterns. *Computational Statistics and Data Analysis in press*.

Takai, D. and P. A. Jones (2002). Comprehensive analysis of CpG islands in human chromosomes 21 and 22. *Proceedings of the National Academies of Science U S A. 99*, 3740–5.

Viterbi, A. (1967). Error bounds for convolutions codes and an asymptotically optimal decoding algorithm. *IEEE Transactions on Information Theory 13*, 260–269.

A Bayesian Method for Non-Gaussian Autoregressive Quantile Function Time Series Models

Yuzhi Cai *
University of Plymouth
Plymouth, PL4 8AA, UK
ycai@plymouth.ac.uk

Bayesian method has been used widely in many areas. Gilks et al (1996) and Berg (2004) have given extensive discussions on these issues. Different types of Markov chain Monte Carlo (MCMC) methods have been proposed to deal with different problems. For example, Green (1995) proposed a reversible jump MCMC method to allow proposals that change the dimensionality of the space; Ball et al. (1999) proposed an MCMC method for hidden Markov processes with applications to ion channel gating mechanism. All the above work is trying to model the distribution of random variables of interest.

On the other hand, modelling the quantiles of random variables of interest is becoming more and more popular. Generally speaking, there are two different types of approaches to modelling quantiles of a time series y_t conditional on $y_1, \ldots, y_{t-1}$. One is the semi-parametric approach proposed by Koenker (2005). His model says that the τ^{th} conditional quantile of y_t given $\mathbf{y}_{t-1} = (y_1, \ldots, y_{t-1})$ is given by

$$q^{\tau}_{y_t|\mathbf{y}_{t-1}} = a^{\tau}_0 + a^{\tau}_1 y_{t-1} + \cdots + a^{\tau}_k y_{t-k},$$

where the parameters are estimated by minimizing the following cost function

$$\min_{\boldsymbol{\beta}} \sum_{t=k+1}^{n} \rho_{\tau}(u_t),$$

where $\boldsymbol{\beta} = (a^{\tau}_0, \ldots, a^{\tau}_k)$ is the parameter vector, $\rho_{\tau}(u_t) = u_t(\tau - I_{[u_t<0]})$, and

$$u_t = y_t - a^{\tau}_0 - a^{\tau}_1 y_{t-1} - \cdots - a^{\tau}_k y_{t-k}$$

for $t = k+1, \ldots, n$. Different methods have been proposed to solve the above optimization problem, see for example, Koenker and D'Orey (1987, 1994). Yu and Moyeed (2001) and Chen and Yu (2008) also proposed a Bayesian method to estimate the model parameters in the case when the data are independent with each other. Cai(2007) and Cai and Stander (2008) extended the Bayesian approach to deal with quantile self-exciting autoregressive time series models.

Note that we say the above approach is a semi-parametric approach because the error term of the model is not specified. Gilchrist (2000) proposed a parametric approach to modelling autoregressive quantile function time series models. An autoregressive quantile function time series model is defined by

$$Q_{y_t}(\tau \mid \mathbf{y}_{t-1}) = a_0 + a_1 y_{t-1} + \cdots + a_k y_{t-k} + \eta Q(\tau, \boldsymbol{\gamma}), \tag{1}$$

where a_i, $i = 0, \ldots, k$, η and $\boldsymbol{\gamma}$ are the model parameters, and $Q(\tau, \boldsymbol{\gamma})$ is the quantile function of the error term with parameter $\boldsymbol{\gamma}$. When $Q(\tau, \boldsymbol{\gamma})$ is not the quantile function of a normal random variable, (1) defines a non-Gaussian quantile function time series model. Model (1) is parametric because we assume that the mathematical form of $Q(\tau, \boldsymbol{\gamma})$ is known.

Gilchrist (2000) also discussed several methods for estimating the parameters of such models. However, to the author's knowledge, no work in the literature can be found on a Bayesian approach to quantile function modelling for time series, which motivated our current research.

It is worth mentioning that the choice of $Q(\tau, \boldsymbol{\gamma})$ is very flexible. For example, the sum of quantile functions gives a new quantile function, the product of two positive quantile functions is also a

*address for correspondence: Dr Yuzhi Cai, School of Mathematics and Statistics, University of Plymouth, Plymouth PL4 8AA, United Kingdom.

quantile function etc. The properties of quantile functions enable us to construct proper statistical models for an observed time series, and to deal with non-Gaussian time series very easily. In this talk, we will consider a simple non-Gaussian quantile function time series model where the error term follows an exponential distribution, we will present an MCMC method to estimate model parameters, we will carry out simulation studies to investigate the performance of the method, and we will also apply the methodology developed to two real time series. We will see that such a quantile function approach to time series modelling indeed provides a very flexible way to study non-Gaussian time series.

Acknowledgments

The author would like to thank for the invitation and the financial support from Isaac Newton Institute for Mathematical Sciences which made it possible for me to present the paper at the workshop.

References

[1] Ball, Frank, Cai, Yuzhi, Kadane, J. B. and O.Hagan, A. (1999). Bayesian inference for ion channel gating mechanisms directly from single channel recordings, using Markov chain Monte Carlo. *Proceedings of the Royal society of London Series A - Mathematical Physical and Engineering Sciences*, 455, 2879-2932.

[2] Berg, Bernd A (2004). Markov Chain Monte Carlo Simulations and Their Statistical Analysis. Singapore, World Scientific.

[3] Cai, Yuzhi and Stander, Julian (2008). Quantile Self-exciting Threshold Autoregressive Time Series Models. Journal of Time Series Analysis. Vol.29, 186–202.

[4] Chen, L. and Yu, K. (2008), Automatic Bayesian Quantile Regression Curve, Statistics and Computing, accepted.

[5] Cai, Yuzhi (2007). A Quantile Approach to US GNP. *Economic Modelling*, 24, 969-979.

[6] Gilchrist, W.G. (2000). Statistical Modelling with Quantile Functions. Chapman & Hall/CRC.

[7] Green, P.J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82, 711732.

[8] Gilks, WR, Richardson, S and Spiegelhalter, DJ (1996). Markov Chain Monte Carlo in Practice. Chapman & Hall/CRC.

[9] Koenker, R. (2005). *Quantile Regression*. Cambridge University Press.

[10] Koenker, R. and D'Orey, V. (1987). Computing regression quantiles. *Applied Statistics*, 36, 383–393.

[11] Koenker, R. and D'Orey, V. (1994). A remark on Algorithm AS229: Computing dual regression quantiles and regression rank scores. *Applied Statistics*, 43, 410–414.

[12] Yu, K. and Moyeed, R.A. (2001). Bayesian quantile regression. *Statistics and Probability Letters*, 54, 437–447.

A Modern Perspective on Auxiliary Particle Filters

Adam M. Johansen
Department of Mathematics
University of Bristol
adam.johansen@bristol.ac.uk

Nick Whiteley
Department of Engineering
University of Cambridge
npw24@cam.ac.uk

Abstract

The auxiliary particle filter (APF) is a popular algorithm for the Monte Carlo approximation of the optimal filtering equations of state space models. This paper presents a summary of several recent developments which affect the practical implementation of this algorithm as well as simplifying its theoretical analysis. In particular, an interpretation of the APF, which makes use of an auxiliary sequence of distributions, allows the approach to be extended to more general Sequential Monte Carlo algorithms. The same interpretation allows existing theoretical results for standard particle filters to be applied directly. Several non-standard implementations and applications are also discussed.

1 Background

1.1 State Space Models

State space models (SSMs, and the closely related hidden Markov models) are very popular statistical models for time series. Such models describe the trajectory of some system of interest as an unobserved E-valued Markov chain, known as the *signal process*, which for the sake of simplicity is treated as being time-homogeneous in this paper. Let $X_1 \sim \nu$ and $X_n|(X_{n-1} = x_{n-1}) \sim f(\cdot|x_{n-1})$ and assume that a sequence of observations, $\{Y_n\}_{n\in\mathbb{N}}$ are available. If Y_n is, conditional upon X_n, independent of the remainder of the observation and signal processes, with $Y_n|(X_n = x_n) \sim g(\cdot|x_n)$, then this describes an SSM.

For any sequence $\{z_n\}_{n\in\mathbb{N}}$, we define $z_{i:j} = (z_i, z_{i+1}, ..., z_j)$. In numerous applications, we are interested in estimating recursively in time an analytically intractable sequence of posterior distributions $\{p(x_{1:n}|y_{1:n})\}_{n\in\mathbb{N}}$, of the form:

$$p(x_{1:n}|y_{1:n}) \propto \nu(x_1)g(y_1|x_1)\prod_{j=2}^{n} f(x_j|x_{j-1})g(y_j|x_j). \tag{1}$$

A great deal has been written about inference for such models – see [1, 2] for example – especially *filtering*, which corresponds to computing the final time marginal of (1) at each time. This article is concerned with a class of Monte Carlo algorithms which address this problem by approximating the distributions of interest with a set of weighted samples. The remainder of this section introduces two standard approaches to this problem, sequential importance resampling (SIR) and the auxiliary particle filter (APF). Section 2 illustrates the strong connection between these algorithms, and provides some guidance upon implementation of the APF. Section 3 then illustrates a number of extensions which are suggested by these connections.

1.2 Sequential Importance Resampling

SIR is one of the most popular techniques for performing inference in SSMs. This technique propagates a collection of weighted samples, termed *particles*, from one iteration to the next in such a way that they provide an approximation of the filtering distribution at each iteration. In fact, as illustrated in algorithm 1, this technique can be used to sample from essentially any sequence of distributions defined on a sequence of spaces of strictly increasing dimension. At its nth iteration, algorithm 1

provides an approximation of $\pi_n(x_{1:n})$. A crucial step in this algorithm is resampling. This involves duplicating particles with high weights and discarding particles with low weights and reweighting to preserve the distribution targeted by the weighted sample. The simplest scheme, multinomial resampling, achieves this by drawing N times from the empirical distribution of the weighted particle set (lower variance alternatives are summarised in [2]).

Algorithm 1 The Generic SIR Algorithm

At time 1
  for $i = 1$ to N **do**
    $X_1^{(i)} \sim q_1(\cdot)$
    $W_1^{(i)} \propto \frac{\pi_1(X_1^{(i)})}{q_1(X_1^{(i)})}$
  end for
  Resample $\left\{X_1^{(i)}, W_1^{(i)}\right\}$ to obtain $\left\{X_1^{\prime(i)}, \frac{1}{N}\right\}$
At time $n \geq 2$
  for $i = 1$ to N **do**
    Set $X_{1:n-1}^{(i)} = X_{1:n-1}^{\prime(i)}$
    Sample $X_n^{(i)} \sim q_n(\cdot|X_{n-1}^{(i)})$
    Set $W_n^{(i)} \propto \frac{\pi_n(X_{1:n}^{(i)})}{q_n(X_n^{(i)}|X_{n-1}^{(i)})\pi_{n-1}(X_{1:n-1}^{(i)})}$
  end for
  Resample $\left\{X_{1:n}^{(i)}, W_n^{(i)}\right\}$ to obtain $\left\{X_{1:n}^{\prime(i)}, \frac{1}{N}\right\}$

In a filtering context, $\pi_n(x_{1:n}) = p(x_{1:n}|y_{1:n})$ and the expectation of some test function φ_n with respect to the filtering distribution, $\overline{\varphi}_n = \int \varphi_n(x_n)p(x_n|y_{1:n})dx_n$ can be estimated using

$$\widehat{\varphi}_{n,SIR}^N = \sum_{i=1}^N W_n^{(i)}\varphi_n(X_n^{(i)})$$

where $W_n^{(i)} = w_n(X_{n-1:n}^{(i)}) \Big/ \sum_{j=1}^N w_n(X_{n-1:n}^{(i)})$ and

$$w_n(x_{n-1:n}) = \frac{\pi_n(x_{1:n})}{q_n(x_n|x_{n-1})\pi_{n-1}(x_{1:n-1})} \propto \frac{g(y_n|x_n)f(x_n|x_{n-1})}{q_n(x_n|x_{n-1})}. \tag{2}$$

Note that (2) depends only on the two most recent components of the particle trajectory, so the corresponding algorithm can be implemented with storage requirements which do not increase over time and is suitable for online applications. In fact, SIR can be viewed as a selection-mutation (genetic-type) algorithm constructed with a precise probabilistic interpretation. Viewing SIR as a particle approximation of a Feynman-Kac flow [3] allows many theoretical results to be established.

1.3 Auxiliary Particle Filters

It is natural to ask whether it is possible to employ knowledge about the next observation *before* resampling to ensure that particles which are likely to be compatible with that observation have a good chance of surviving – is it possible to preserve diversity in the particle set by taking into account the immediate future as well as the present when carrying out selection? The APF first proposed by [4, 5] invoked an auxiliary variable construction in answer to this question.

The essence of this APF was that the sampling step could be modified to sample an auxiliary variable, corresponding to a particle index, according to a distribution which weights each particle in terms of it compatibility with the coming observation. A suitable weighting is provided by some $\widehat{p}(y_n|x_{n-1})$, an approximation of $\int g(y_n|x_n)f(x_n|x_{n-1})dx_n$ (if the latter is not available analytically). It is straightforward to see that this is equivalent to resampling according to those weights before carrying out a standard sampling and resampling iteration. A similar approach in which the auxiliary weights are combined with those of the standard weighting was proposed in [6], which involved a single resampling during each iteration of the algorithm. See algorithm 2.

Algorithm 2 Auxiliary Particle Filter

At time 1
  for $i = 1$ to N **do**
    $X_1^{(i)} \sim q_1(\cdot)$
    $\widetilde{W}_1^{(i)} \propto \frac{g(y_1|X_1^{(i)})\nu(X_1^{(i)})}{q_1(X_1^{(i)})}$
  end for
At time $n \geq 2$
  for $i = 1$ to N **do**
    Set $W_{n-1}^{(i)} \propto \widetilde{W}_{n-1}^{(i)} \times \widehat{p}(y_n|X_{n-1}^{(i)})$
  end for
  Resample $\left\{X_{n-1}^{(i)}, W_{n-1}^{(i)}\right\}$ to obtain $\left\{X_{n-1}^{\prime(i)}, \frac{1}{N}\right\}$
  for $i = 1$ to N **do**
    Set $X_{n-1}^{(i)} = X_{n-1}^{\prime(i)}$
    Sample $X_n^{(i)} \sim q_n(\cdot|X_{n-1}^{(i)})$
    Set $\widetilde{W}_n^{(i)} \propto \frac{g(y_n|X_n^{(i)})f(X_n^{(i)}|X_{n-1}^{(i)})}{\widehat{p}(y_n|X_{n-1}^{(i)})q_n(X_n^{(i)}|X_{n-1}^{(i)})}$
  end for

2 Interpretation and Implementation

Whilst the APF has seen widespread use, remarkably the first asymptotic analyses of the algorithm have appeared very recently. These analyses provide some significant insights into the performance of the algorithm and emphasize some requirements that a successful implementation must meet.

2.1 The APF as SIR

When one considers the APF as a sequence of weighting and sampling operations it becomes apparent that it also has an interpretation as a mutation-selection algorithm. In fact, with a little consideration it is possible to interpret the APF as being an SIR algorithm.

It was noted in [7] that the APF described in [6] corresponds to the SIR algorithm which is obtained by setting

$$\pi_n(x_{1:n}) = \widehat{p}(x_{1:n}|y_{1:n+1}) \propto p(x_{1:n}|y_{1:n})\widehat{p}(y_{n+1}|x_n). \tag{3}$$

In the SIR interpretation of the APF $p(x_{1:n}|y_{1:n})$ is not approximated directly, but rather importance sampling is used to estimate $\overline{\varphi}_n$, with the importance distribution $\pi_{n-1}(x_{1:n-1})q_n(x_n|x_{n-1})$. The resulting estimate is given by

$$\widehat{\varphi}_{n,APF}^N = \sum_{i=1}^N \widetilde{W}_n^{(i)} \varphi_n(X_n^{(i)}) \tag{4}$$

where $\widetilde{W}_n^{(i)} = \widetilde{w}_n(X_{n-1:n}^{(i)}) \Big/ \sum_{j=1}^N \widetilde{w}_n(X_{n-1:n}^{(j)})$ and

$$\widetilde{w}_n(x_{n-1:n}) = \frac{p(x_{1:n}|y_{1:n})}{\pi_{n-1}(x_{1:n-1})q_n(x_n|x_{n-1})} \propto \frac{g(y_n|x_n)f(x_n|x_{n-1})}{\widehat{p}(y_n|x_{n-1})q_n(x_n|x_{n-1})}. \tag{5}$$

Only the case in which resampling is carried out once per iteration has been considered here. Empirically this case has been preferred for many years and one would intuitively expect it to lead to lower variance estimates. However, it would be straightforward to apply the same reasoning to the scenario in which resampling is carried out both before and after auxiliary weighting as in the original implementations (doing this leads to an SIR algorithm with twice as many distributions as previously but there is no difficulty in constructing such an algorithm).

One of the principle advantages of identifying the APF as a particular type of SIR algorithm is that many detailed theoretical results are available for the latter class of algorithm. Indeed, many of the results provided in [3], for example, can be applied directly to the APF via this interpretation. Thus formal convergence results can be obtained (see [7] for a central limit theorem and some discussion of other results which follow directly) without any additional analysis.

2.2 Implications for Implementation

From an implementation point of view, perhaps the most significant feature of this interpretation is that it makes clear the criticality of choosing a $\widehat{p}(y_n|x_{n-1})$ which is *more* diffuse than $p(y_n|x_{n-1})$ (as a function of x_{n-1}). For importance sampling schemes in general, it is well known that a proposal distribution with lighter tails than the target distribution can lead to an estimator with infinite variance. In the case of the APF the proposal distribution is defined in terms of $\widehat{p}(y_n|x_{n-1})$. It is therefore clear that the popular choice of approximating the predictive likelihood by the likelihood evaluated at the mode of the transition density is a dangerous strategy. This is likely to explain the poor-performance of APF algorithms based on this idea which have appeared in the literature. One simple option is to take

$$\widehat{p}(y_n|x_{n-1}) \propto \int \widehat{g}(y_n|x_n)\widehat{f}(x_n|x_{n-1})dx_n$$

with the approximations to the likelihood and transition densities being chosen to have heavier tails than the true densities and to permit this integral to be evaluated.

Whilst it remains sensible to attempt to approximate the optimal (in the sense of minimising the variance of the importance weights) transition density $q_n(x_n|x_{n-1}) \propto f(x_n|x_{n-1})g(y_n|x_n)$ and the true predictive likelihood, it is not the case that the APF necessarily out-performs the SIR algorithm using the same proposal even in this setting. This phenomenon is related to the fact that the mechanism by which samples are proposed at the current iteration of the algorithm impacts the variance of estimates made at subsequent time steps. This is immediately apparent from the asymptotic variance expressions which can be found together with an illustrative example in [7].

2.3 Direct Analysis

A direct analysis of the particle system underlying the APF was performed recently [8]. This confirmed the intuitive and empirical results that resampling once per iteration leads to a lower variance estimate than resampling twice. One principle component of this work was the determination of the auxiliary weighting function which minimises the variance of estimates of a particular test function obtained *one step ahead* of the current iterations. Whilst this is of some theoretical interest, it would seem to be necessary to exercise some caution when attempting to use such a weighting function in practice, as it could have a deleterious effect upon estimates of the integral of that *same function* at *later* iterations.

If the test function of interest can be estimated without obtaining a good characterisation of the entire distribution (for example, if that function has a support which is substantially smaller than that of the distribution) then it may be desirable to concentrate the entire particle set in a small region to minimise the immediate variance, but this could lead to severe problems at subsequent iterations. It is for precisely the same reason that the use of customised proposal distributions tuned for a specific test function are not generally used in particle filtering and thus a more conservative approach, with less adaptation in the proposal mechanism remains sensible.

3 Applications and Extensions

The innovation of the APF is essentially that in sampling from a sequence of distributions using a SIR strategy, it can be advantageous to employ an auxiliary sequence of distributions which take account of one-step-ahead knowledge about the distributions of interest and to use importance sampling to provide estimates under those distributions. This section summarises some other applications of this principle outside of the filtering domain in which it has previously been applied.

3.1 (Auxiliary) Sequential Monte Carlo Samplers

SMC Samplers are a class of algorithms for sampling iteratively from a sequence of distributions, denoted by $\{\pi_n(x_n)\}_{n\in\mathbb{N}}$, defined upon a sequence of potentially arbitrary spaces, $\{E_n\}_{n\in\mathbb{N}}$, [9]. The approach involves the application of SIR to a cleverly constructed sequence of synthetic distributions which admit the distributions of interest as marginals. It is consequently straightforward to employ the same strategy as that used by the APF – see [10] which also illustrates that convergence

results for this class of algorithms follow directly. In this context it is not always clear that there is a good choice of auxiliary distributions, although it is relatively natural in some settings.

The synthetic distributions are $\widetilde{\pi}_n(x_{1:n}) = \pi_n(x_n)\prod_{p=1}^{n-1} L_p(x_{p+1}, x_p)$, where $\{L_n\}_{n\in\mathbb{N}}$ is a sequence of 'backward' Markov kernels *from* E_n *into* E_{n-1}. With this structure, an importance sample from $\widetilde{\pi}_n$ is obtained by taking the path $x_{1:n-1}$, a sample from $\widetilde{\pi}_{n-1}$, and extending it with a Markov kernel, K_n, which acts from E_{n-1} into E_n, providing samples from $\widetilde{\pi}_{n-1} \times K_n$ and leading to the importance weight:

$$w_n(x_{n-1:n}) = \frac{\widetilde{\pi}_n(x_{1:n})}{\widetilde{\pi}_{n-1}(x_{1:n-1})K_n(x_{n-1}, x_n)} = \frac{\pi_n(x_n)L_{n-1}(x_n, x_{n-1})}{\pi_{n-1}(x_{n-1})K_n(x_{n-1}, x_n)}. \tag{6}$$

In many applications, each $\pi_n(x_n)$ can only be evaluated pointwise, up to a normalizing constant and the importance weights defined by (6) are normalised in the same manner as in the SIR algorithm. Resampling may then be performed.

If one wishes to sample from a sequence of distributions $\{\pi_n\}_{n\in\mathbb{N}}$ then an alternative to directly implementing an SMC sampler which targets this sequence of distributions, is to employ an auxiliary sequence of distributions, $\{\mu_n\}_{n\in\mathbb{N}}$ and an importance sampling correction (with weights $\widetilde{w}_n(x_n) = \pi_n(x_n)/\mu_n(x_n)$) to provide estimates. This is very much in the spirit of the APF. Such a strategy was termed auxiliary SMC (ASMC) in [10]. Like the APF, the objective is to maintain a more diverse particle set by using information before resampling rather than after.

3.1.1 Resample-Move: Inverting Sampling and Resampling

As has been previously noted, [9], in a setting in which one has a fixed state space, $E_n = E$ at every iteration, and employs a MCMC kernel of invariant distribution π_n as the proposal, and makes use of the auxiliary kernel:

$$L_{n-1}(x_n, x_{n-1}) = \frac{\pi_n(x_{n-1})K_n(x_{n-1}, x_n)}{\pi_n(x_n)},$$

the importance weights are simply $w_n(x_{n-1}, x_n) = \pi_n(x_{n-1})/\pi_{n-1}(x_{n-1})$ which is independent of the proposed state, x_n.

Consequently, it is intuitively clear that one should resample *before* proposing new states in the interests of maximising sample diversity. This has been observed previously, for example by [11]. Indeed doing so leads to algorithms with the same structure as the Resample-Move (RM) particle filtering algorithm [12]. By making the following identifications, it is possible to cast this approach into the form of an ASMC sampler.

$$\begin{aligned}
\mu_n(x_n) &= \pi_{n+1}(x_n) \\
L_{n-1}(x_n, x_{n-1}) &= \frac{\mu_{n-1}(x_{n-1})K_n(x_{n-1}, x_n)}{\mu_{n-1}(x_n)} = \frac{\pi_n(x_{n-1})K_n(x_{n-1}, x_n)}{\pi_n(x_n)} \\
w_n(x_{n-1:n}) &= \frac{\mu_n(x_n)}{\mu_{n-1}(x_n)} = \frac{\pi_{n+1}(x_n)}{\pi_n(x_n)} \\
\widetilde{w}_n(x_n) &= \mu_{n-1}(x_n)/\mu_n(x_n) = \pi_n(x_n)/\pi_{n+1}(x_n).
\end{aligned}$$

This allows existing theoretical results to be applied to both RM and its generalisations.

3.1.2 Filtering Piecewise-Deterministic Processes

As an example, the SMC Samplers framework was employed by [13] to provide filtering estimates for a class of continuous-time processes. This also illustrates that SMC samplers and their auxiliary counterparts can provide useful extensions of SIR-type algorithms in time-series analysis. Piecewise-Deterministic Processes (PDP's) are a class of stochastic processes whose sample paths, $\{\zeta_t\}_{t\geq 0}$ evolve deterministically in continuous time between a sequence of random times $\{\tau_j\}_{j\in\mathbb{N}}$, at which the path jumps to new, random values $\{\theta_j\}_{j\in\mathbb{N}}$. Filtering for partially observed PDP models involves computing a sequence of posterior distributions given observations

$\{Y_n\}_{n\in\mathbb{N}}$, where $Y_n = H(\zeta_{t_n}, V_n)$, V_n is a noise disturbance and $\{t_n\}_{n\in\mathbb{N}}$ is an increasing sequence of observation times. The nth such posterior $\pi_n(k_n, \theta_{n,0:k_n}, \tau_{n,1;k_n}|y_{1:n})$, is a distribution over $E_n = \biguplus_{k=0}^{\infty}\{k\} \times \Theta^{k+1} \times \mathbb{T}_{n,k}$,where $\Theta \subset \mathbb{R}^d$ is a parameter space, $\mathbb{T}_{n,k} = \{\tau_{n,1:k_n} : 0 \leq \tau_{n,1} < ... < \tau_{n,k_n} \leq t_n\}$. The posterior distribution is specified by

$$\pi_n(k_n, \theta_{n,0:k_n}, \tau_{n,1;k_n}|y_{1:n}) \propto \nu(\theta_{n,0})S(t_n, \tau_{n,k_n}) \prod_{j=2}^{k_n} f(\theta_{n,j}, \tau_{n,j}|\theta_{n,j-1}, \tau_{n,j-1}) \prod_{p=1}^{n} g(y_n|\zeta_{t_n}),$$

with the convention $\tau_{n,0} = 0$ and where $S(t_n, \tau_{n,k_n})$ is the survivor function associated with the prior distribution on inter-jump times for the interval $[0, t_n]$. The SMC Samplers framework is applied to approximate the distributions of interest, using a proposal kernel consisting of a mixture of moves which extend each particle from E_{n-1} to E_n by adjusting recent jump-time/parameter pairs and adding new ones. An auxiliary scheme for filtering can be obtained by selecting the auxiliary distribution μ_n to be:

$$\mu_n(k_n, \theta_{n,0:k_n}, \tau_{n,1;k_n}) \propto V_n(\theta_{n,k_n}, \tau_{n,k_n})\pi_n(k_n, \theta_{n,0:k_n}, \tau_{n,1;k_n}|y_{1:n}),$$

where $V_n(\theta_{n,k_n}, \tau_{n,k_n})$ is a non-negative potential function which provides information about y_{n+1}. This strategy was seen to perform well in [13].

3.2 The Probability Hypothesis Density Filter

Multi-object tracking involves the online estimation of the time-varying number and positions of a collection of hidden objects, given a sequence of noisy observations. In principle, filtering for a multi-object tracking model involves computing a sequence of distributions with the same form as (1). Here, E is $E = \biguplus_{k=0}^{\infty} \mathcal{X}^k$, with $\mathcal{X} \subset \mathbb{R}^d$ is the state-space of an individual object: each $X_n = X_{n,1:k_n}$ is actually a random number, k_n, of points, each in $\mathcal{X}$, and can be regarded as a *spatial point process* [14]. The observation set at time n, $Y_n = Y_{n,1:m_n}$, is defined similarly. Performing filtering on such spaces is practically very difficult due to the high and variable dimensionality. The Probability Hypothesis Density (PHD) Filter, [15], approximates the optimal filter for this problem by assuming that the state process is Poisson process a-posteriori and characterising the intensity of that process, α.

For ease of presentation, we here consider a specific tracking model for which the PHD recursion has the following prediction/update structure at its nth iteration:

$$\alpha_n(x_n) = \int_{\mathcal{X}} f(x_n|x_{n-1})p_S(x_{n-1})\breve{\alpha}_{n-1}(x_{n-1})\mathrm{d}x_{n-1} + \gamma(x_n), \tag{7}$$

$$\breve{\alpha}_n(x_n) = \sum_{p=1}^{m_n} \frac{g(y_{n,p}|x_n)}{\mathcal{Z}_{n,p}}\alpha_n(x_n), \tag{8}$$

where for $p = 1, 2, ..., m_n$, $\mathcal{Z}_{n,p} = \int_E g(y_{n,p}|x)\alpha_n(x)dx + \kappa(y_{n,p})$. In this notation, $\alpha_n(x)$ and $\breve{\alpha}_n(x)$ are respectively termed the predicted and updated intensities at time n, $\gamma(x)$ is the intensity of new objects, $p_S(x)$ is the survival probability, $f(x_n|x_{n-1})$ is the transition kernel of an individual object, and $\kappa(y)$ is the intensity of the clutter. We denote by $g(y_{n,p}|x)$ and m_n the likelihood for the pth observation and the total number of observations at iteration n respectively.

SMC methods may employed to approximate the sequence of intensity functions $\{\breve{\alpha}_n(x_n)\}_{n\in\mathbb{N}}$. In contrast to the case of particle filters which approximate probability distributions, it is necessary for the collection of weighted samples used here to characterise the total mass of the intensity function in addition to its form. Akin to the APF, an auxiliary SMC implementation (and references to other approaches) can be found in [16], which demonstrates that this recursion is particularly well suited to this approach , which outperforms more direct particle implementations.

In outline, this approach introduces an extended state space $\mathcal{X}' = \mathcal{X} \cup \{s\}$, where s is an isolated "source" point which does not belong to $\mathcal{X}$. Then define an intensity function denoted $\beta_n(x_{n-1:n})$ on $\mathcal{X} \times \mathcal{X}'$ as follows:

$$\beta_n(x_{n-1:n}) = \sum_{p=1}^{m_n} \frac{g(y_{n,p}|x_n)}{\mathcal{Z}_{n,p}} \left[f(x_n|x_{n-1})p_S(x_{n-1})\breve{\alpha}_{n-1}(x_{n-1})\mathbb{I}_{\mathcal{X}}(x_{n-1}) + \gamma(x_n)\delta_s(x_{n-1})\right]. \tag{9}$$

Note that $\beta_n(x_{n-1:n})$ admits $\breve{\alpha}_n(x_n)$ under integration. The algorithm of [16] effectively approximates each term in (9) separately. Assume that there is available a particle approximation of $\breve{\alpha}_{n-1}(x_{n-1})$. Then, for each $p \in \{1, 2, ..., m_n\}$, a mechanism which weights and resamples particles is executed with target distribution $\pi_{n-1,p}(x_{n-1})$ on $\mathcal{X}'$, where:

$$\pi_{n-1,p}(x_{n-1}) \propto \widehat{p}(y_{n,p}|x_{n-1}) \left[\breve{\alpha}_{n-1}(x_{n-1})\mathbb{I}_{\mathcal{X}}(x_{n-1}) + \delta_s(x_{n-1})\right],$$

$\widehat{p}(y_{n,p}|x_{n-1})$ being an approximation of $p(y_{n,p}|x_{n-1})$, which is itself defined by

$$p(y_{n,p}|x_{n-1}) = \mathbb{I}_{\mathcal{X}}(x_{n-1}) \int_{\mathcal{X}} g(y_{n,p}|x_n) f(x_n|x_{n-1}) dx_n + \mathbb{I}_{\{s\}}(x_{n-1}) \int_{\mathcal{X}} g(y_{n,p}|x_n)\gamma(x_n) dx_n.$$

Proposals are then made from a kernel $q_{n,p}(x_n|x_{n-1})$. The importance weight which yields a particle approximation to the pth term in (9) is given by:

$$\widetilde{w}_{n,p}(x_{n-1:n}) \propto \frac{g(y_{n,p}|x_n)[f(x_n|x_{n-1})p_S(x_{n-1})\mathbb{I}_{\mathcal{X}}(x_{n-1}) + \gamma(x_n)\mathbb{I}_{\{s\}}(x_{n-1})]}{q_{n,p}(x_n|x_{n-1})\pi_{n-1,p}(x_{n-1})},$$

Each normalizing constant $\mathcal{Z}_{n,p}$ is also estimated by IS, much as in SMC algorithms for SSMs. The particle sets are then pooled, yielding a particle approximation of $\breve{\alpha}_n(x_n)$. [16] also provides the optimal number of particles to assign to each term in (9).

3.3 Further Stratifying the APF

It is common knowledge that the use of multinomial resampling in a particle filter unnecessarily increases the Monte Carlo variance of the associated estimators and that the use of systematic or stratified approaches can significantly reduce that variance. The APF is, of course, no exception and one should always employ minimum variance resampling strategies. Under some circumstances it may be possible to introduce some further stratification in the APF.

Consider again the SSM from section 1.1. Let $(A_p)_{p=1}^M$ denote a partition of E. Introducing an auxiliary stratum-indicator variable, $m_n = \sum_{p=1}^M p\mathbb{I}_{A_p}(x_n)$, we redefine the SSM on a higher dimensional space, with the signal process being $E \times \{1, 2, ..., M\}$-valued, with transition kernel:

$$r(x_n, m_n|x_{n-1}, m_{n-1}) = r(x_n|m_n, x_{n-1})r(m_n|x_{n-1}),$$

where:

$$r(x_n|m_n, x_{n-1}) \propto \mathbb{I}_{A_{m_n}}(x_n)f(x_n|x_{n-1}), \qquad r(m_n|x_{n-1}) = \int_{A_{m_n}} f(x_n|x_{n-1})dx_n.$$

The initial distribution of the extended chain is defined in a similar manner and the likelihood remains essentially unchanged. The posterior distributions for the extended model then obey the following recursion:

$$p(x_{1:n}, m_{1:n}|y_{1:n}) \propto g(y_n|x_n)r(x_n|m_n, x_{n-1})r(m_n|x_{n-1})p(x_{1:n-1}, m_{1:n-1}|y_{1:n-1}). \quad (10)$$

Note that the marginal distribution of $x_{1:n}$ in (10) coincides with the original model.

As in the SIR interpretation of the APF, we then construct an auxiliary sequence of distributions, $\{\pi(x_{1:n-1}, m_{1:n})\}_{n\in\mathbb{N}}$, which will be targeted with an SIR algorithm, where:

$$\pi(x_{1:n-1}, m_{1:n}) \propto \widehat{p}(y_n|m_n, x_{n-1})\widehat{r}(m_n|x_{n-1})p(x_{1:n-1}, m_{1:n-1}|y_{1:n-1}). \quad (11)$$

For each i, we first draw each $X_n^{(i)}|x_{n-1}^{(i)}, m_n^{(i)} \sim q(\cdot|x_{n-1}^{(i)}, m_n^{(i)})$. Then, instead of randomly sampling a value $m_{n+1}^{(i)}$, we evaluate one importance weight for every possible value of m_{n+1}, resulting in a collection of $N \times M$ weighted sample points. The resampling step of the SIR algorithm then draws N times from the resulting distribution on $\{1, 2, ..., N\} \times \{1, 2, ..., M\}$. A related method has been proposed in the context of tracking problems, but without exploiting the benefits of stratification via low variance resampling [17].

The importance weight which targets $p(x_{1:n}, m_{1:n}|y_{1:n})$ (i.e. the analogue of (5)) is then:

$$\widetilde{w}_n(x_{n-1:n}, m_n) \propto \frac{g(y_n|x_n)f(x_n|x_{n-1})}{\widehat{p}(y_n|m_n, x_{n-1})\widehat{r}(m_n|x_{n-1})q_n(x_n|m_n, x_{n-1})}.$$

This effectively assigns both a parent particle *and* a stratum to each offspring.This approach may be of interest in the context of *switching* SSMs, where the state space has a natural partition structure by definition.

4 Conclusions

This paper has summarised the state of the art of the auxiliary particle filter. Our intention is to provide some insight into the behaviour of the APF and its relationship with other particle-filtering algorithms, in addition to summarising a number of recent methodological extensions. The most significant point is perhaps this: the APF is simply an example of a sequential estimation situation in which one can benefit (by introducing information about subsequent distributions earlier) from approximating the *wrong* sequence of distributions and using an importance sampling correction. Other such scenarios exist and the same approach can be used when addressing them.

References

[1] O. Cappé, E. Moulines, and T. Ryden. *Inference in Hidden Markov Models*. Springer Verlag, New York, 2005.

[2] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Statistics for Engineering and Information Science. Springer Verlag, New York, 2001.

[3] P. Del Moral. *Feynman-Kac formulae: genealogical and interacting particle systems with applications*. Probability and Its Applications. Springer Verlag, New York, 2004.

[4] M. K. Pitt and N. Shephard. Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association*, 94(446):590–599, 1999.

[5] M. K. Pitt and N. Shephard. Auxiliary variable based particle filters. In Doucet et al. [2], chapter 13, pages 273–293.

[6] J. Carpenter, P. Clifford, and P. Fearnhead. An improved particle filter for non-linear problems. *IEEE Proceedings on Radar, Sonar and Navigation*, 146(1):2–7, 1999.

[7] A. M. Johansen and A. Doucet. A note on the auxiliary particle filter. *Statistics and Probability Letters*, 2008. To appear.

[8] R. Douc, E. Moulines, and J. Olsson. On the auxiliary particle filter. Technical Report 0709.3448v1, arXiv, September 2007.

[9] P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo methods for Bayesian Computation. In *Bayesian Statistics 8*. Oxford University Press, 2006.

[10] A. M. Johansen and A. Doucet. Auxiliary variable sequential Monte Carlo methods. Technical Report 07:09, University of Bristol, Department of Mathematics – Statistics Group, University Walk, Bristol, BS8 1TW, UK, July 2007.

[11] A. M. Johansen, A. Doucet, and M. Davy. Particle methods for maximum likelihood parameter estimation in latent variable models. *Statistics and Computing*, 18(1):47–57, March 2008.

[12] W. R. Gilks and Carlo Berzuini. RESAMPLE-MOVE filtering with Cross-Model jumps. In Doucet et al. [2], pages 117–138.

[13] N. Whiteley, A. M. Johansen, and S. Godsill. Monte Carlo filtering of piecewise-deterministic processes. Technical Report CUED/F-INFENG/TR-592, University of Cambridge, Department of Engineering, Trumpington Street, Cambridge, CB2 1PZ, 2007.

[14] S.S. Singh, B.-N. Vo, A. Baddeley, and S. Zuyev. Filters for spatial point processes. Technical Report CUED/F-INFENG/TR-591, University of Cambridge, Department of Engineering, Trumpington Street, Cambridge,CB1 2PZ, United Kingdom, 2007.

[15] R. P. S. Mahler. An introduction to multisource-multitarget statistics and its applications. Technical monograph, Lockheed Martin, March 2000.

[16] N. Whiteley, S. Singh, and S. Godsill. Auxiliary particle implementation of the probability hypothesis density filter. Technical Report CUED F-INFENG/590, University of Cambridge, Department of Engineering, Trumpington Street, Cambridge,CB1 2PZ, United Kingdom, 2007.

[17] R. Karlsson and N. Bergman. Auxiliary particle filters for tracking a maneuvering target. In *Proceedings of the 39th IEEE Conference on Decision and Control*, pages 3891–3895, December 2000.

State Estimation in High Dimensional Systems: The Method of The Ensemble Unscented Kalman Filter

Xiaodong Luo
Mathematical Institute, University of Oxford
24-29 St Giles', Oxford, UK, OX1 3LB
luox@maths.ox.ac.uk

Irene M. Moroz
Mathematical Institute, University of Oxford
24-29 St Giles', Oxford, UK, OX1 3LB
moroz@maths.ox.ac.uk

Abstract

The ensemble Kalman filter (EnKF) is a Monte Carlo implementation of the Kalman filter, which is often adopted to reduce the computational cost when dealing with high dimensional systems. In this work, we propose a new EnKF scheme based on the concept of the unscented transform [12], which therefore will be called the ensemble unscented Kalman filter (EnUKF). Under the assumption of Gaussian distribution of the estimation errors, it can be shown analytically that, the EnUKF can achieve more accurate estimations of the ensemble mean and covariance than the ordinary EnKF. Therefore incorporating the unscented transform into an EnKF may benefit its performance. Numerical experiments conducted on a 40-dimensional system [14] support this argument.

1 Introduction

The Kalman filter (KF) is a recursive data processing algorithm [15]. It optimally estimates the states of linear stochastic systems that are driven by Gaussian noise, and are observed through linear observation operators, which possibly also suffer from additive Gaussian errors. However, if there exists nonlinearity from either the dynamical systems or the observation operators, or, if neither the dynamical noise nor the observational noise follows any Gaussian distribution, then the Kalman filter becomes suboptimal. To tackle the problems of nonlinearity and non-Gaussianity, there are some strategies one may employ. For example, to handle the problem of nonlinearity, one may expand the nonlinear function locally in a Taylor series and keep the expansion terms only up to second order. This leads to the extended Kalman filter (EKF) (e.g., [4]). To deal with the problem of non-Gaussianity, one may specify a Gaussian mixture model (GMM) to approximate the underlying probability density function (pdf), such that the KF algorithm is applicable to the individual distributions of the GMM [18]. More generally, one may adopt the sequential Monte Carlo method (also known as the particle filter, e.g., [19]), which utilizes the empirical pdf obtained from a number of particles to represent the true pdf, wherein the problems of both nonlinearity and non-Gaussianity are taken into account during the pdf approximation.

For practical large-scale problems like weather forecasting, the computational cost is another issue of great concern. In such circumstances, direct application of the KF or EKF scheme is prohibitive because of the computational cost of evolving the full covariance matrix forward. While for the particle filter, because of its slow convergence rate, the required number of the particles for proper approximations may be well above many thousands for even low dimensional nonlinear systems [11]. For the sake of computational efficiency, the so-called ensemble Kalman filter (EnKF) was proposed in [5]. It is essentially a Monte Carlo implementation of the Kalman filter. Under the framework of the EnKF, the computational cost can be significantly reduced.

In this paper we will introduce a modified framework of the EnKF incorporating the concept of the unscented transform [9, 10, 12], which therefore will be called the ensemble unscented KF (EnUKF for short). Under the assumption that the estimation errors follow a Gaussian distribution, it can be shown that the EnUKF has better accuracies in estimating the ensemble mean and covariance than the ordinary EnKF. To save space, here we omit the analytic results and provide the numerical comparison only.

This paper is organized as follows. We firstly review the framework of the ordinary EnKF in section 2. We then proceed to introduce the EnUKF in section 3. In section 4, we use the nonlinear model in [14] to demonstrate the performance of the EnUKF, and compare it to the ordinary EnKF. Finally we discuss and conclude the work in section 5.

2 The framework of the EnKF

For simplicity in illustration, we consider the perfect model scenario. Suppose that we have a perfect m-dimensional discrete dynamical system

$$\mathbf{x}_{k+1} = \mathcal{M}_{k,k+1}\left(\mathbf{x}_k\right), \tag{1}$$

where $\mathbf{x}_k$ denotes the m-dimensional system state at the instant k, and $\mathcal{M}_{k,k+1}$ is the transition operator mapping $\mathbf{x}_k$ to $\mathbf{x}_{k+1}$. We also assume that the system is observed through a p-dimensional observer $\mathcal{H}_k$ such that

$$\mathbf{y}_k = \mathcal{H}_k\left(\mathbf{x}_k\right) + \mathbf{v}_k, \tag{2}$$

where $\mathbf{y}_k$ are the p-dimensional observations at instant k, and $\mathbf{v}_k$, the p-dimensional observation errors, are independent of the model state, and follow a Gaussian process with zero mean and covariance $\mathbf{R}_k$.

For convenience of discussion, let us first introduce some concepts customarily used in the community of data assimilation and meteorology. A state at instant k, which is propagated from a state at the previous instant $k-1$, is called the *background* at the assimilation cycle k, and usually denoted by $\mathbf{x}_k^b$. Given the observations $\mathbf{y}_k$, one introduces a correction to $\mathbf{x}_k^b$ based on the KF algorithm, and obtains an updated state, which is called the *analysis*, and usually denoted by $\mathbf{x}_k^a$. One then propagates $\mathbf{x}_k^a$ forward again to obtain the background $\mathbf{x}_{k+1}^b$ at the next cycle. In this way, one can apply the KF algorithm for state estimation recursively. We call the update from the background to the analysis the filtering step, and the propagation from the analysis to the background at the next cycle the propagation step.

2.1 The filtering step

Without lost of generality, one may assume that there is an n-member ensemble of the analysis $\{\mathbf{x}_{k-1,i}^a : i = 1, 2\cdots, n\}$ available at the end of the $(k-1)$-th assimilation cycle. So at the filtering step of the k-th cycle, a propagated ensemble

$$\mathbf{X}_k^b = \left\{\mathbf{x}_{k,i}^b : \mathbf{x}_{k,i}^b = \mathcal{M}_{k,k+1}\left(\mathbf{x}_{k-1,i}^a\right), i = 1, 2, \cdots, n\right\}$$

can be obtained. The sample mean $\hat{\mathbf{x}}_k^b$ and covariance $\hat{\mathbf{P}}_k^b$ can be evaluated according to the following unbiased estimators [1].

$$\hat{\mathbf{x}}_k^b = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_{k,i}^b; \quad \hat{\mathbf{P}}_k^b = \frac{1}{n-1}\sum_{i=1}^{n}\left(\mathbf{x}_{k,i}^b - \hat{\mathbf{x}}_k^b\right)\left(\mathbf{x}_{k,i}^b - \hat{\mathbf{x}}_k^b\right)^T. \tag{3}$$

In practice, the approximation covariance $\hat{\mathbf{P}}_k^b$ need not be calculated. Instead, it is customary to compute

$$\begin{aligned}\hat{\mathbf{P}}_{xh}^k =& \frac{1}{n-1}\sum_{i=1}^{n}\left(\mathbf{x}_{k,i}^b - \hat{\mathbf{x}}_k^b\right)\left(\mathcal{H}_k\left(\mathbf{x}_{k,i}^b\right) - \mathcal{H}_k\left(\hat{\mathbf{x}}_k^b\right)\right)^T,\\ \hat{\mathbf{P}}_{hh}^k =& \frac{1}{n-1}\sum_{i=1}^{n}\left(\mathcal{H}_k\left(\mathbf{x}_{k,i}^b\right) - \mathcal{H}_k\left(\hat{\mathbf{x}}_k^b\right)\right)\left(\mathcal{H}_k\left(\mathbf{x}_{k,i}^b\right) - \mathcal{H}_k\left(\hat{\mathbf{x}}_k^b\right)\right)^T.\end{aligned} \tag{4}$$

The Kalman gain $\mathbf{K}_k$ can be obtained through

$$\mathbf{K}_k = \hat{\mathbf{P}}_{xh}^k\left(\hat{\mathbf{P}}_{hh}^k + \mathbf{R}_k\right)^{-1}. \tag{5}$$

[1] In some works, e.g., [20], the authors may choose other estimators.

With additional information from the observations, one can update the background ensemble according to a certain scheme. As an example, we consider the ensemble transform Kalman filter (ETKF) [3, 20]. In this scheme, on one hand, the ensemble mean $\hat{\mathbf{x}}_k^a$ is updated as follows

$$\hat{\mathbf{x}}_k^a = \hat{\mathbf{x}}_k^b + \mathbf{K}_k \left(\mathbf{y}_k - \mathcal{H} \left(\hat{\mathbf{x}}_k^b \right) \right). \tag{6}$$

On the other, let

$$\delta \mathbf{X}_k^b = \left[\mathbf{x}_{k,i}^b - \hat{\mathbf{x}}_k^b, \cdots, \mathbf{x}_{k,n}^b - \hat{\mathbf{x}}_k^b \right], \; i = 1, \cdots, n$$

be the matrix consisting of the perturbations of the background, then the perturbations of the analysis

$$\delta \mathbf{X}_k^a = \left[\delta \mathbf{x}_{k,i}^a, \cdots, \delta \mathbf{x}_{k,n}^a \right], \; i = 1, \cdots, n$$

are updated from $\delta \mathbf{X}_k^b$ according to

$$\delta \mathbf{X}_k^a = \delta \mathbf{X}_k^b \, \mathbf{T}, \tag{7}$$

where $\mathbf{T}$ is the transformation matrix derived in [3]. Given $\hat{\mathbf{x}}_k^a$ and $\delta \mathbf{X}_k^a$, the ensemble of the analysis $\mathbf{X}_k^a = \left\{ \mathbf{x}_{k,i}^a : i = 1, 2, \cdots, n \right\}$ is generated according to

$$\mathbf{x}_{k,i}^a = \hat{\mathbf{x}}_k^a + \delta \mathbf{x}_{k,i}^a, \; i = 1, 2, \cdots, n. \tag{8}$$

The ensemble mean $\hat{\mathbf{x}}_k^a$ is already given in Eq. (6), while the ensemble covariance $\hat{\mathbf{P}}_k^a$ is computed by

$$\hat{\mathbf{P}}_k^a = \delta \mathbf{X}_k^a \left(\delta \mathbf{X}_k^a \right)^T / (n-1). \tag{9}$$

2.2 The propagation step

After the updates, one propagates each member of the analysis ensemble through the system model so as to obtain the background ensemble at the next assimilation cycle.

3 The ensemble unscented Kalman filter

For consistency, we again take Eqs. (1) and (2) as the m-dimensional system model and the p-dimensional observer respectively. Moreover, we also assume that there exists a set of system states $\{\mathcal{X}_{k-1,i}^a, i = 0, 1, \cdots, 2l_{k-1}\}$, called the sigma points, at the $(k-1)$-th step. Correspondingly, we denote the set of the propagated sigma points at the k-th cycle by $\{\mathcal{X}_{k,i}^b : \mathcal{X}_{k,i}^b = \mathcal{M}_{k-1,k} \left(\mathcal{X}_{k-1,i}^a \right), i = 0, 1, \cdots, 2l_{k-1}\}$, which are associated with a set of weights $\left\{ W_{k-1,i}, \cdots, W_{k-1,l_{k-1}} \right\}$ specified according to Eq. (16). The weighted sample mean and covariance of the background at the k-th cycle are given by

$$\hat{\mathbf{x}}_k^b = \sum_{i=0}^{2l_{k-1}} W_{k-1,i} \, \mathcal{X}_{k,i}^b, \tag{10a}$$

$$\hat{\mathbf{P}}_k^b = \sum_{i=0}^{2l_{k-1}} W_{k-1,i} \left(\mathcal{X}_{k,i}^b - \hat{\mathbf{x}}_k^b \right) \left(\mathcal{X}_{k,i}^b - \hat{\mathbf{x}}_k^b \right)^T + \beta \left(\mathcal{X}_{k,0}^b - \hat{\mathbf{x}}_k^b \right) \left(\mathcal{X}_{k,0}^b - \hat{\mathbf{x}}_k^b \right)^T, \tag{10b}$$

where the second term on the rhs of Eq. (10b) is introduced to reduce the approximation error. In the case that $\mathbf{x}$ follows a Gaussian distribution, the choice of $\beta = 2$ is shown to be optimal [9].

The above evaluation scheme is also applicable to the projection of the background ensemble such that

$$\begin{aligned} \hat{\mathbf{P}}_{xh}^k = & \sum_{i=0}^{2l_{k-1}} W_{k-1,i} \left(\mathcal{X}_{k,i}^b - \hat{\mathbf{x}}_k^b \right) \left(\mathcal{H}_k \left(\mathcal{X}_{k,i}^b \right) - \mathcal{H}_k \left(\hat{\mathbf{x}}_k^b \right) \right)^T \\ & + \beta \left(\mathcal{X}_{k,0}^b - \hat{\mathbf{x}}_k^b \right) \left(\mathcal{H}_k \left(\mathcal{X}_{k,0}^b \right) - \mathcal{H}_k \left(\hat{\mathbf{x}}_k^b \right) \right)^T, \\ \hat{\mathbf{P}}_{hh}^k = & \sum_{i=0}^{2l_{k-1}} W_{k-1,i} \left(\mathcal{H}_k \left(\mathcal{X}_{k,i}^b \right) - \mathcal{H}_k \left(\hat{\mathbf{x}}_k^b \right) \right) \left(\mathcal{H}_k \left(\mathcal{X}_{k,i}^b \right) - \mathcal{H}_k \left(\hat{\mathbf{x}}_k^b \right) \right)^T \\ & + \beta \left(\mathcal{H}_k \left(\mathcal{X}_{k,0}^b \right) - \mathcal{H}_k \left(\hat{\mathbf{x}}_k^b \right) \right) \left(\mathcal{H}_k \left(\mathcal{X}_{k,0}^b \right) - \mathcal{H}_k \left(\hat{\mathbf{x}}_k^b \right) \right)^T. \end{aligned} \tag{11}$$

Then, following Eq. (5), the Kalman gain is

$$\mathbf{K}_k = \hat{\mathbf{P}}_{xh}^k \left(\hat{\mathbf{P}}_{hh}^k + \mathbf{R}_k \right)^{-1}. \tag{5}$$

With the above information, the mean and covariance of the analysis can be computed according to

$$\hat{\mathbf{x}}_k^a = \hat{\mathbf{x}}_k^b + \mathbf{K}_k \left(\mathbf{y}_k - \mathcal{H}_k \left(\hat{\mathbf{x}}_k^b \right) \right), \tag{12a}$$

$$\hat{\mathbf{P}}_k^a = \hat{\mathbf{P}}_k^b - \mathbf{K}_k \left(\hat{\mathbf{P}}_{xh}^k \right)^T. \tag{12b}$$

Apart from obtaining the updated sample mean and covariance, we also aim to generate a set of sigma points as the analysis ensemble, which will then be propagated to the next assimilation cycle. For this purpose, one may consider using an existing EnKF scheme, for example, the ETKF. However, in order to avoid doubling the ensemble size at the $(k+1)$-th cycle, some sigma points have to be discarded. To do this, the sample mean can be preserved by maintaining the symmetry about $\hat{\mathbf{x}}_k^a$ among the remaining sigma points, while the corresponding sample covariance, denoted by $\tilde{\mathbf{P}}_k^a$, can only be an approximation to $\hat{\mathbf{P}}_k^a$. This may appear to be a complicated problem for the existing EnKF schemes to design a selection criterion, because the perturbations produced by them have no indications of the relative importance for covariance approximation.

To tackle the above problem, the truncated singular value decomposition (TSVD) [7] is adopted in this work. Suppose that $\hat{\mathbf{P}}_k^a$ can be expressed as

$$\hat{\mathbf{P}}_k^a = \mathbf{E}_k \mathbf{D}_K \left(\mathbf{E}_k \right)^T, \tag{13}$$

where $\mathbf{D}_K = \text{diag}(\sigma_{k,1}^2, \cdots, \sigma_{k,m}^2)$ is a diagonal matrix consisting of the eigenvalues of $\hat{\mathbf{P}}_k^a$, which are sorted in descending order, i.e., $\sigma_{k,i}^2 \geq \sigma_{k,j}^2 \geq 0$ for $i > j$; and $\mathbf{E}_K = [\mathbf{e}_{k,1}, \cdots, \mathbf{e}_{k,m}]$ is the matrix consisting of the corresponding eigenvectors. Then, one can produce l_k perturbations, in terms of the first l_k vectors of $\sigma_{k,i}\mathbf{e}_{k,i}$, and add them to the sample mean $\hat{\mathbf{x}}_k^a$ to form l_k sigma points. Another symmetric l_k sigma points can also be produced by subtracting the perturbations from the sample mean. Overall, the above procedure can be summarized as follows

$$\begin{aligned}
\mathcal{X}_{k,0}^a &= \hat{\mathbf{x}}_k^a, \\
\mathcal{X}_{k,i}^a &= \hat{\mathbf{x}}_k^a + (l_k + \lambda)^{1/2} \sigma_{k,i} \mathbf{e}_{k,i}, \; i = 1, \cdots, l_k, \\
\mathcal{X}_{k,i}^a &= \hat{\mathbf{x}}_k^a - (l_k + \lambda)^{1/2} \sigma_{k,i-l} \mathbf{e}_{k,i-l}, \; i = l_k + 1, \cdots, 2l_k,
\end{aligned} \tag{14}$$

where λ is an adjustable scaling parameter. For convenience, we will hereafter call l_k the truncation number.

In our implementation, we let l_k be an integer such that

$$\begin{aligned}
\sigma_{k,i}^2 &> \text{trace} \left(\hat{\mathbf{P}}_k^a \right) / h_k \,, i = 1, \cdots, l_k \\
\sigma_{k,i}^2 &\leq \text{trace} \left(\hat{\mathbf{P}}_k^a \right) / h_k \,, i > l_k + 1
\end{aligned} \tag{15}$$

where h_k is the threshold at the k-th cycle. Moreover, to prevent l_k being too large or too small, we also specify a lower bound l_l and an upper bound l_u. One may need to adjust the threshold h_k at each cycle to let $l_l \leq l_k \leq l_u$.

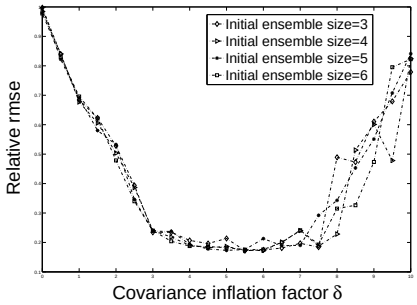
After the sigma points are generated, a set of corresponding weights can be specified as follows

$$W_{k,0} = \frac{\lambda}{l_k + \lambda}; \quad W_{k,i} = \frac{1}{2(l_k + \lambda)}, \; i = 1, \cdots, 2l_k. \tag{16}$$

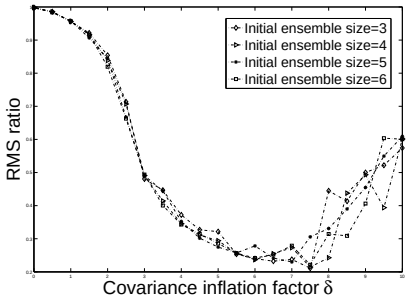
Finally, all the sigma points in Eq. (14), which are associated with a set of weights given by Eq. (16), are propagated forward to the next assimilation cycle.

We adopt the time averaged relative rms error (relative rmse for short) to measure the performance of the EnUKF, which is defined as

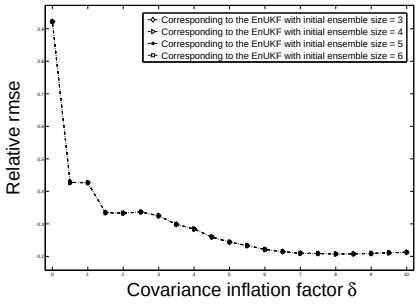
$$e_r = \frac{1}{k_{max}} \sum_{k=1}^{k_{max}} \| \hat{\mathbf{x}}_k^a - \mathbf{x}_k^{tr} \|_2 / \| \mathbf{x}_k^{tr} \|_2, \tag{17}$$



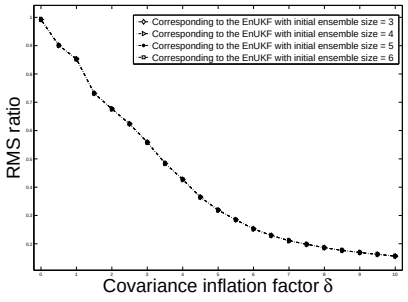
(a) Relative rmse of the EnUKF vs the covariance inflation factor δ



(b) RMS ratio of the EnUKF vs the covariance inflation factor δ



(c) Relative rmse of the ETKF with the ensemble size equal to 13



(d) RMS ratio of the ETKF with the ensemble size equal to 13

Figure 1: Effects of the covariance inflation factor δ on the performance of the EnUKF.

where k_{max} is the maximum assimilation cycle, $\mathbf{x}_k^{tr}$ denotes the true state at the k-th cycle, and $\|\bullet\|_2$ means the L_2 norm. Moreover, we also use the time averaged rms ratio to examine the similarity of the truth to the sigma points, which also qualitatively reflects the performance in estimating the error covariance, e.g., overestimation or underestimation (cf. [1, 21] and the references therein). By definition, the time averaged rms ratio, denoted by R, is computed as follows

$$R = \frac{1}{k_{max}} \sum_{k=1}^{k_{max}} (2l_k + 1) \, \|\hat{\mathbf{x}}_k^a - \mathbf{x}_k^{tr}\|_2 / \sum_{i=0}^{2l_k} \|\mathcal{X}_{k,i}^a - \mathbf{x}_k^{tr}\|_2 \, . \tag{18}$$

If the true state is statistically indistinguishable from the sigma points, then the expectation of R is

$$R_e = \sqrt{(l_{eff} + 1)/(2l_{eff} + 1)},$$

where l_{eff} is the "effective" truncation number over the whole assimilation window. Note that $R_e \approx 0.71$ for any large l_{eff}, so for simplicity we let l_{eff} equal the average of the truncation number $\bar{l}$, i.e., $l_{eff} = \bar{l} = \sum_{i=1}^{k_{max}} l_k / k_{max}$. $R > R_e$ means that the covariance of the sigma points underestimates the error of state estimation, while $R < R_e$ implies the opposite, i.e., overestimation of the error of state estimation [16, 21].

4 Numerical experiments with a 40-dimensional system

This section is dedicated to examining the performance of the EnUKF, and comparing it with one of the prevailing ensemble Kalman filters in the community of data assimilation: the ETKF. We choose the m-dimensional system model due to Lorenz and Emanuel [13, 14] (LE98 model hereafter) as the testbed. The LE98 model is a simplified system used to model atmospheric dynamics, and "shares certain properties with many atmospheric models" [14]. We consider the perfect model scenario,

wherein the governing equations are described as follows

$$\frac{dx_i}{dt} = (x_{i+1} - x_{i-2})\, x_{i-1} - x_i + F,\ i = 1, \cdots, m. \tag{19}$$

The quadratic terms simulate the advection, the linear term represents the internal dissipation, while the constant F acts as the external forcing ([13]). The variables x_i's are defined cyclically such that $x_{-1} = x_{m-1}$, $x_0 = x_m$, and $x_{m+1} = x_1$.

We choose the observer $\mathcal{H}_k$ to be a time-invariant identity operator. Specifically, given a system state $\mathbf{x}_k = [x_{k,1}, \cdots, x_{k,m}]^T$ at the k-th assimilation cycle, the observations are obtained according to

$$\mathbf{y}_k = \mathcal{H}_k(\mathbf{x}_k) + \mathbf{v}_k = \mathbf{x}_k + \mathbf{v}_k\,, \tag{20}$$

where $\mathbf{v}_k$ follows an m-dimensional Gaussian distribution $N(\mathbf{0}, \mathbf{R}_k)$ with the covariance matrix $\mathbf{R}_k$ being the $m \times m$ identity matrix $\mathbf{I}_m$.

In our experiments, we set $m = 40$ and $F = 8$ and integrate the system through the fourth-order Runge-Kutta method. We choose the length of the integration window to be 100 dimensionless units, and the integration time step to be 0.05 units (corresponding to about a 6-h interval in reality, see [14]), thus there are 2000 assimilation cycles overall.

In order to improve the performances of filters, we also consider two additional techniques. One is the method of covariance inflation, which is based on the observation that the covariance of the analysis error will be systematically underestimated in the EnKF [21]. Therefore, it may be beneficial to increase either the background error covariance before updating the background, or the analysis error covariance after the updating [2, 17, 21]. In this work, we follow the method used in [2, 21] and choose to multiply the perturbations to the sample mean $\mathbf{x}_k^a$ of the analysis by a constant $1 + \delta$, which is equivalent to increasing the analysis error covariance by a factor $(1 + \delta)^2$.

The other technique is the covariance filter [6, 8], which introduces the Schur-product to a covariance matrix in order to reduce the effect of sample errors. We say a length scale of covariance filter is optimal within a certain range if it minimizes the relative rmse among all possible values. Numerical simulations (not reported here) show that for the EnUKF, the optimal length scale $l_c = 200$, while for the ETKF, its optimal length scale $l_c = 240$. For simplicity, we choose $l_c = 240$ for both filters.

For the purpose of comparison, we randomly select an initial condition $\mathbf{x}_1$, and use it to start a control run. The observations $\mathbf{Y} = \{\mathbf{y}_k\}_{k=1}^{2000}$ are obtained by adding Gaussian white noise to the states of the control run at each cycle, in accordance with Eq. (20). In subsequent simulations, both the EnUKF and the ETKF will start with the same initial condition $\mathbf{x}_1$, and use the same observations $\mathbf{Y}$ for assimilation. To initialize both the filters, at the first assimilation cycle we randomly generate a background ensemble $\mathbf{X}_1^b = \{\mathbf{x}_{1,i} : i = 1, \cdots, n\}$. Given $\mathbf{X}_1^b$ and $\mathbf{x}_1$, the ETKF is already able to start running recursively. For the EnUKF, however, at the first cycle there is no propagated sigma points from the previous cycle. But, similar to the ETKF, one may use the background ensemble $\mathbf{X}_1^b$ to compute the sample mean and covariance of the analysis, and then generate the sigma points accordingly. After propagating the sigma points forward, the EnUKF can start running recursively from the second cycle.

For the EnUKF, we let parameters $\beta = 2$, $\lambda = -2$, the threshold $h_1 = 1000$, the lower bound $l_l = 3$, the upper bound $l_u = 6$, the length scale of covariance filter $l_c = 240$, and the covariance inflation factor δ vary from 0 to 10, with an even increment of 0.5. We consider the scenarios with different ensemble sizes $n = 3, 4, 5, 6$ at the first assimilation cycle in order to explore the effect of initial ensemble size on the performance of the EnUKF. The corresponding relative rms error and ratio, as functions of the covariance inflation factor, are plotted in Figs. 1(a) and 1(b) respectively.

From the above two figures, it can be seen that different initial ensemble sizes $n = 3, 4, 5, 6$ leads to similar behaviors of both the relative rmse and rms ratio. Interestingly, a larger initial ensemble size does not necessarily guarantee a smaller rmse error. This can be observed either from Fig. 1(a) by fixing the covariance inflation factor δ at some point, say $\delta = 1.5$, or from Table 1 by comparing the minimum relative rms errors.

Fig. 1(a) shows that, as the covariance inflation factor δ increases from 0, the relative rmse of the EnUKF tends to decline. However, if δ already gets too large, say $\delta > 6$, then further increments in δ will instead boost the relative rmse. An examination on the rms ratio also reveals the same trend,

Table 1: Minima of the relative rms errors in Figs. 1(a) and 1(c).

Ensemble Filter	Minimum of the relative rms errors			
	n=3	n=4	n=5	n=6
EnUKF	0.1719	0.1722	0.1730	0.1753
ETKF	0.2074	0.2074	0.2074	0.2074

although, as indicated in Fig. 1(b), the turning points, now at $\delta = 7.5$ for $n = 3, 4, 6$ and $\delta = 7$ for $n = 5$, are larger than those of the relative rms errors. To make the sigma point indistinguishable from the truth (i.e., rms ratio ≈ 0.71), one needs an inflation factor $\delta \approx 2.5$. However, modestly larger inflation factors, say, $2.5 < \delta < 6$, can benefit the performance of the EnUKF in terms of the relative rmse, although they also cause the overestimation of the error covariance.

For the ETKF, we let the length scale l_c of the covariance filter and the covariance inflation factor δ be the same as those in the EnUKF. Suppose that in a run of the EnUKF we have the average truncation number $\bar{l}$. Then for comparison, we consider the ETKF with an ensemble size $n = \text{ceil}(2\bar{l} + 1)$, where $\text{ceil}(s)$ means the nearest integer that is larger than, or equal to, the real number s. In our experiments, the EnUKF with different initial ensemble sizes $n = 3, 4, 5, 6$ leads to the same value $\text{ceil}(2\bar{l} + 1) = 13$, so it is not suprising to find in Figs. 1(c) and 1(d) that the relative rms errors and ratios of the ETKF, which correspond to the EnUKF starting with different initial ensemble sizes, actually coincide.

From Fig. 1(c), one can see that, starting from $\delta = 0$, as the covariance inflation factor increases, the relative rmse of the ETKF tends to decrease. However, unlike the situation in the EnUKF, in the test range, as δ gets larger, say $\delta > 7$, the corresponding relative rmse enters a plateau region. The rms ratio indicates a similar behaviour. As δ increases, the change of the rms ratio becomes smaller, or in other words, the curve appears more and more flat. In order to make the ensemble of the ETKF indistinguishable from the truth, one needs the inflation factor $\delta \approx 1.5$. Like the EnUKF, overestimation of the error covariance (i.e., $\delta > 1.5$) can also benefit the performance of the ETKF in terms of the relative error.

We use the minimum relative rms errors of the EnUKF and the ETKF to compare their performances. To this end, in Table 1 we list the minimum relative rms errors of the EnUKF with different initial ensemble sizes, and the corresponding values of the ETKF with the ensemble size about twice the average truncation number plus 1. Note that different initial ensemble sizes $n = 3, 4, 5, 6$ in the EnUKF lead to the same ensemble size of the ETKF. Therefore, in Table 1, the ETKF has the same minimum relative rmse even in different columns. For the EnUKF with different initial ensemble sizes, the minimum relative rms errors are roughly 0.17, while for the ETKF, the minimum relative rmse is approximately 0.20. In this sense, the EnUKF outperforms the ETKF.

5 Conclusions

We proposed a new ensemble Kalman filter scheme based on the concept of the unscented transform. We introduced some modifications in order to make the unscented transform suitable for large-scale problems. In a more lengthy paper (in preparation), we show that, under the assumption of Gaussian distribution of the estimation error, the EnUKT has better accuracies in estimating the sample mean and covariance than the ordinary EnKF, e.g., the ETKF. Therefore incorporating the unscented transform into an EnKF may benefit its performance. Numerical simulations reported in this work support our arguments.

References

[1] Anderson, J. L. (2001). An ensemble adjustment Kalman filter for data assimilation. *Mon. Wea. Rev.*, 129, pp. 2884–2903.

[2] Anderson, J. L. & Anderson, S. L. (1999). A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Mon. Wea. Rev.*, 127, pp. 2741–2758.

[3] Bishop, C. H., Etherton, B. J. & Majumdar, S. J. (2001). Adaptive sampling with ensemble transform Kalman filter. Part I: theoretical aspects. *Mon. Wea. Rev.*, 129, pp. 420–436.

[4] Evensen, G. (1992). Using the extended Kalman filter with a multilayer quasi-geostrophic ocean model. *J. Geophys. Res.*, 97, pp. 17,905–17,924.

[5] Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.*, 99(C5), pp. 10,143–10,162.

[6] Hamill, T. M., Whitaker, J. S. & Snyder, C. (2001). Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter. *Mon. Wea. Rev.*, 129, pp. 2776–2790.

[7] Hansen, P. C. (1987). The truncated SVD as a method for regularization. *BIT*, 27, pp. 534 – 553.

[8] Houtekamer, P. L. & Mitchell, H. L. (2001). A sequential ensemble kalman filter for atmospheric data assimilation. *Mon. Wea. Rev.*, 129, pp. 123–137.

[9] Julier, S. J. (2004). The scaled unscented transformation, in *The Proceedings of the American Control Conference*, Anchorage, AK, pp. 4555 – 4559.

[10] Julier, S. J. & Uhlmann, J. K. (1996). *A general method for approximating nonlinear transformations of probability distributions*, *Tech. rep.*, Department of Engineering Science, University of Oxford.

[11] Julier, S. J. & Uhlmann, J. K. (2002). Reduced Sigma Point Filters for the Propagation of Means and Covariances Through Nonlinear Transformations, in *The Proceedings of the American Control Conference*, Anchorage, AK, pp. 887–892.

[12] Julier, S. J., Uhlmann, J. K. & Durrant-Whyte, H. F. (1995). A new approach for filtering nonlinear systems, in *The Proceedings of the American Control Conference*, Seattle, Washington, pp. 1628–1632.

[13] Lorenz, E. N. (1996). Predictability-a problem solved, in *Predictability.*, ed. Palmer, T., ECMWF, Reading, UK.

[14] Lorenz, E. N. & Emanuel, K. A. (1998). Optimal sites for supplementary weather observations: Simulation with a small model. *J. Atmos. Sci.*, 55, pp. 399–414.

[15] Maybeck, P. (1979). *Stochastic Models, Estimation, and Control*, Academic Press.

[16] Murphy, J. M. (1988). The impact of ensemble forecasts on predictability. *Quart. J. Roy. Meteor. Soc.*, 114, pp. 463 – 493.

[17] Ott, E. et al. (2004). A local ensemble Kalman filter for atmospheric data assimilation. *Tellus*, 56A, pp. 415–428.

[18] Smith, K. W. (2007). Cluster ensemble Kalman filter. *Tellus*, 59A, pp. 749–757.

[19] van Leeuwen, P. J. (2003). A variance minimizing filter for large-scale applications. *Mon. Wea. Rev.*, 131, pp. 2071–2084.

[20] Wang, X., Bishop, C. H. & Julier, S. J. (2004). What's better, an ensemble of positive-negative pairs or a centered simplex ensemble. *Mon. Wea. Rev.*, 132, pp. 1590–1605.

[21] Whitaker, J. S. & Hamill, T. M. (2002). Ensemble data assimilation without perturbed observations. *Mon. Wea. Rev.*, 130, pp. 1913–1924.

Clustering of Time Course Gene-Expression Data via Mixture Regression Models

Geoffrey J. McLachlan
Department of Mathematics & Institute for Moelcular Bioscience,
University of Queensland
Department of Mathematics, University of Queensland,
Brisbane 4072, Australia
gjm@maths.uq.edu.au

S.K. Ng
School of Medicine, Griffith University
University Drive, Meadowbrook QLD 4131, Australia
s.ng@griffith.edu.au

Kui Wang
Department of Mathematics, University of Queensland,
Brisbane 4072, Australia
kwang@maths.uq.edu.au

Abstract

In this paper, we consider the use of mixtures of linear mixed models to cluster data which may be correlated and replicated and which may have covariates. This approach can thus be used to cluster time series data. For each cluster, a regression model is adopted to incorporate the covariates, and the correlation and replication structure in the data are specified by the inclusion of random effects terms. The procedure is illustrated in its application to the clustering of time-course gene expression data.

1 Introduction

Finite mixture models are being commonly used in a wide range of applications in practice concerning density estimation and clustering; see, for example, McLachlan and Peel [1]. We let $\boldsymbol{Y}$ denote a random vector consisting of p feature variables associated with the random phenomenon of interest. We let $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$ denote an observed random sample of size n on $\boldsymbol{Y}$. With the finite mixture model-based approach to density estimation and clustering, the density of $\boldsymbol{Y}$ is modelled as a mixture of a number (g) of component densities $f_i(\boldsymbol{y})$ in some unknown proportions $\pi_1, \ldots, \pi_g$. That is, each data point is taken to be a realization of the mixture probability density function (p.d.f.),

$$f(\boldsymbol{y}; \boldsymbol{\Psi}) = \sum_{i=1}^{g} \pi_i f_i(\boldsymbol{y}), \tag{1}$$

where the mixing proportions π_i are nonnegative and sum to one. In density estimation, the number of components g can be taken sufficiently large for (1) to provide an arbitrarily accurate estimate of the underlying density function. For clustering purposes, each component in the mixture model (1) corresponds to a cluster. The posterior probability that an observation with feature vector $\boldsymbol{y}_j$ belongs to the ith component of the mixture is given by

$$\tau_i(\boldsymbol{y}_j) = \pi_i f_i(\boldsymbol{y}_j)/f(\boldsymbol{y}_j) \tag{2}$$

for $i = 1, \ldots, g$. A probabilistic clustering of the data into g clusters can be obtained in terms of the fitted posterior probabilities of component membership for the data.

An outright partitioning of the observations into g nonoverlapping clusters $C_1, \ldots, C_g$ is effected by assigning each observation to the component to which it has the highest estimated posterior probability of belonging. Thus the ith cluster C_i contains those observations $\boldsymbol{y}_j$

$$\hat{z}_{ij} = \arg\max_h \hat{\tau}_h(\boldsymbol{y}_j), \tag{3}$$

and $\hat{\tau}_i(\boldsymbol{y}_j)$ is an estimate of $\tau_i(\boldsymbol{y}_j)$. As the notation implies, $\hat{z}_{ij}$ can be viewed as an estimate of z_{ij} which, under the assumption that the observations come from a mixture of g groups $G_1, \ldots, G_g$, is defined to be one or zero according as the jth observation $\boldsymbol{y}_j$ does or does not come from G_i $(i = 1, \ldots, g; j = 1, \ldots, n)$.

In this paper, we wish to focus on the approach proposed by Ng et al. [2] for the clustering of data from time-course microarray experiments, where thousands of genes are assayed repeatedly over several time-points. An example will be given to illustrate its application. Although attention is focussed here solely on the clustering of the gene profiles that can be formed from the output from a series of microarray experiments, the procedure is widely applicable to the clustering of data from other experimental sources.

In the sequel, it is assumed that the observed data vector $\boldsymbol{y}_j$ $(j = 1, \ldots, n)$ contains the expression levels of the jth gene obtained from a series of p microarray experiments; see, for example, McLachlan et al. [3]. Typically in such problems, the number of genes n is very large relative to the number of microarray experiments p. In molecular biology, the $\boldsymbol{y}_j$ are referred to as the gene profiles. The underlying idea for clustering the gene profiles is that if coregulation indicates shared functionality, then clusters defined to this level of abstraction represent biological modules. If the microarray experiments were measured at p different time points, then the problem is one of clustering time-course data (that is, time series data).

2 Mixtures of Linear Mixed Models

We consider the so-called EMMIX-WIRE (**EM**-based **MIX**ture analysis **W**ith **R**andom **E**ffects) procedure developed by Ng et al. [2] to handle the clustering of correlated data over time that may be replicated. They adopted conditionally a mixture of linear mixed models to specify the correlation structure between the variables and to allow for correlations among the observations.

To formulate this procedure, we consider the clustering of n gene profiles $\boldsymbol{y}_j$ $(j = 1, \ldots, n)$, where we let $\boldsymbol{y}_j = (\boldsymbol{y}_{1j}^T, \ldots, \boldsymbol{y}_{mj}^T)^T$ contain the expression values for the jth gene profile and $\boldsymbol{y}_{tj} = (y_{1tj}, \ldots, y_{r_t tj})^T$ $(t = 1, \ldots, m)$ contains the r_t replicated values in the tth biological sample $(t = 1, \ldots, m)$ on the jth gene. The dimension d of $\boldsymbol{y}_j$ is given by $p = \sum_{t=1}^m r_t$. With the EMMIX-WIRE procedure, the observed d-dimensional vectors $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$ are assumed to have come from a mixture of a finite number, say g, of components in some unknown proportions $\pi_1, \ldots, \pi_g$, which sum to one. Conditional on its membership of the ith component of the mixture, the profile vector $\boldsymbol{y}_j$ for the jth gene $(j = 1, \ldots, n)$ follows the model

$$\boldsymbol{y}_j = \boldsymbol{X}\boldsymbol{\beta}_i + \boldsymbol{U}\boldsymbol{b}_{ij} + \boldsymbol{V}\boldsymbol{c}_i + \boldsymbol{\epsilon}_{ij} \quad (i = 1, \ldots, g), \tag{4}$$

where the elements of the q_β-dimensional vector $\boldsymbol{\beta}_i$ are fixed effects (unknown constants) used in modelling the conditional mean of $\boldsymbol{y}_j$ in the ith component $(i = 1, \ldots, g)$. In (4), $\boldsymbol{b}_{ij}$ (a q_b-dimensional vector) and $\boldsymbol{c}_i$ (a q_c-dimensional vector) represent the unobservable gene- and cluster-specific random effects, respectively. These random effects represent the variation due to the heterogeneity of genes and samples (corresponding to $\boldsymbol{b}_i = (\boldsymbol{b}_{i1}^T, \ldots, \boldsymbol{b}_{ip}^T)^T$ and $\boldsymbol{c}_i$, respectively). The random effects $\boldsymbol{b}_i$ and $\boldsymbol{c}_i$, and the measurement error vector $(\boldsymbol{\epsilon}_{i1}^T, \ldots, \boldsymbol{\epsilon}_{ip}^T)^T$ are assumed to be mutually independent, where $\boldsymbol{X}$, $\boldsymbol{U}$, and $\boldsymbol{V}$ are known design matrices of the corresponding fixed or random effects, respectively. If the covariance matrix $\boldsymbol{H}_i$ is taken to be diagonal, then the expression levels on the jth gene in different biological samples are taken to be independent. The presence of the random effect $\boldsymbol{c}_i$ for the expression levels of genes in the ith component induces a correlation between the profiles of genes within the same cluster. This is in contrast to the mixed-effects models approaches in Luan and Li [4], McLachlan et al[3], Celeux et al. [5], and Qin and Self [6] that involve only gene-specific random effects. Their methods thus require the independence assumption for the genes which, however, will not hold in practice for all the genes. Recently, Booth et al. [7]

have adopted a Bayesian approach to this problem in which genes within the same cluster are taken to be correlated.

With the LMM, the distributions of $\boldsymbol{b}_{ij}$ and $\boldsymbol{c}_i$ are taken, respectively, to be multivariate normal $N_{q_b}(\boldsymbol{0}, \boldsymbol{H}_i)$ and $N_{q_c}(\boldsymbol{0}, \theta_{ci}\boldsymbol{I}_{q_c})$, where $\boldsymbol{H}_i$ is a $q_b \times q_b$ covariance matrix. The measurement error vector $\boldsymbol{\epsilon}_{ij}$ is also taken to be multivariate normal $N_p(\boldsymbol{0}, \boldsymbol{A}_i)$, where $\boldsymbol{A}_i = \text{diag}(\boldsymbol{W}\boldsymbol{\xi}_i)$ is a diagonal matrix constructed from the vector $(\boldsymbol{W}\boldsymbol{\xi}_i)$ with $\boldsymbol{\xi}_i = (\sigma^2_{i1}, \ldots, \sigma^2_{iq_e})^T$ and $\boldsymbol{W}$ a known $p \times q_e$ zero-one design matrix.

We now consider an example in which we apply the EMMIX-WIRE procedure to a real data set.

3 Example: Yeast Cell Data

In this example, we consider the CDC28 dataset, which contains more than 6000 genes measured at 17 time points (0, 10, 20, ..., 160) over 160 minutes, which is about two periods of yeast cell under CDC28 condition. Cho et al. [8] and Yeung et al. [9] identified and clustered some of the 6000 genes into different functional groups. For example, Yeung et al. [9] presented 384 genes corresponding to five functional groups, among which there are 237 genes falling into four MIPS functional groups (DNA synthesis and replication, organization of centrosome, nitrogen and sulphur metabolism, and ribosomal proteins). Wong et al. [10] reanalysed the 237 cell cycle data, using their two-stage clustering method and found that it outperformed the other methods that they tried. They were an hierarchical method, k-means, SOM, SOTA, and a normal mixture model-based procedure, which were all used to cluster the 237 genes into $g = 4$ clusters. On comparing the latter with the four MIPS functional groups, they reported that the the Rand Index (RI) for their two-stage method was equal to 0.7087. In this paper, we shall compare the EMMIX-WIRE procedure with the two-stage clustering method.

In this example, the gene profile vector $\boldsymbol{y}_j$ for the jth gene is given by

$$\boldsymbol{y}_j = (y_{j1}, \ldots, y_{jp})^T,$$

where y_{jt} denotes the expression level of the jth gene at time $t\,(t = 1, \ldots, p)$ and p=17. Before proceeding to fit the model (4), we first estimated the period T in the linear regression model in which

$$y_{jt} = \beta_0 + \beta_1 cos(2\pi t/T) + \beta_2 sin(2\pi t/T) + e_{jt},$$

where $t_j = 0, 10, 20, \ldots, 160$, and T is the period, and where it is assumed that $e_{jt} \overset{\text{i.i.d.}}{\sim} N(0, \sigma^2)$. To estimate the period T of the data, we first fixed T as its lower limit T_0, and then calculated the Least Squares (LS) estimate and its mean squared error. We then increased T_0 by 1 to get a new T, and then calculated the LS estimate and its MSE. This was repeated until a reasonable upper limit of T, $T_1 (> T_0)$, was obtained. Comparing all the MSE's, the LS estimate of T corresponding to the minimum MSE is taken as our estimated period T. Using the dataset of 384 genes posted by Yeung et al. [9], we obtained an estimated cell cycle period of 73min, assuming the initial phase to be zero. As a period of 73min is about half of 160min, it would seem to be a reasonable estimate. Also, since the 237 cell cycle data is a subset of the 384 cell cycle data, we assume here that it follows the same time cycle of 73min.

The model (4) was fitted with $\beta_i = (\beta_{1i}, \beta_{2i})^T$ as the fixed-effects vector for the ith component and with the tth row of the design matrix $\boldsymbol{X}$, corresponding to the time point t, given by

$$(cos(2\pi t/T) \qquad sin(2\pi t/T)) \tag{5}$$

for $t = 1, \ldots p$. The design matrix $\boldsymbol{U}$ was taken to be $\boldsymbol{1}_p$ (that is, $q_b = 1$) with $\boldsymbol{b}_{ij} = b_{ij}$, the common random effect for all time points shared by the jth gene, and $\boldsymbol{H}_i = \boldsymbol{I}_p$. The cluster-specific random effect $\boldsymbol{c}_i$ was specified as $\boldsymbol{c}_i = (c_{i1}, \ldots, c_{ip})^T$ with $q_c = p$ and $\boldsymbol{V} = \boldsymbol{I}_p$. With respect to the error terms, we took $\boldsymbol{W} = \boldsymbol{I}_p$ with $q_e = p$.

Concerning the number of components, we report in Table 1 the values of BIC obtained for various levels of the number of components g. As we were unable to calculate the likelihood exactly under the model (4) in the case of nonzero cluster-specific random-effects terms $\boldsymbol{c}_i$, we approximated it by taking the gene-profile vectors to be independently distributed in forming the log likelihood in

Table 1: Values of BIC for Various Levels of the Number of Components g

The	Number	of	Components	
2	3	4	5	6	7
10883	10848	10837	10865	10890	10918

calculating the value of BIC. According to the tabulated values of BIC in Table 1, we should choose $g = 4$ components, which agrees with the number of MIPS functional groups in these genes.

For $g = 4$, we found that the estimated variance $\boldsymbol{\theta}_{ci}$ for the cluster-specific random-effects term was equal to 0.227, 0.280, 0.043, and 0.137, which indicates some level of correlation within at least three of the four clusters. The Rand Index and its adjusted value were equal to 0.7808 and 0.5455, which compare favourably to the corresponding values of 0.7087 and 0.3697, as obtained by Wong et al. [10] for their method. On permuting the cluster labels to minimize the error rate of the clustering with respect to the four MIPS functional groups, we obtained an error rate of 0.291. We also clustered the genes into four clusters by not having cluster-specific random-effects terms $\boldsymbol{c}_i$ in (4), yielding lower values of 0.7152 and 0.4442 for the Rand Index and its adjustment. The estimated error rate was equal to 0.316. Hence in this example, the use of cluster-specific random-effects terms leads to a clustering that corresponds more closely to the underlying functional groups than without their use.

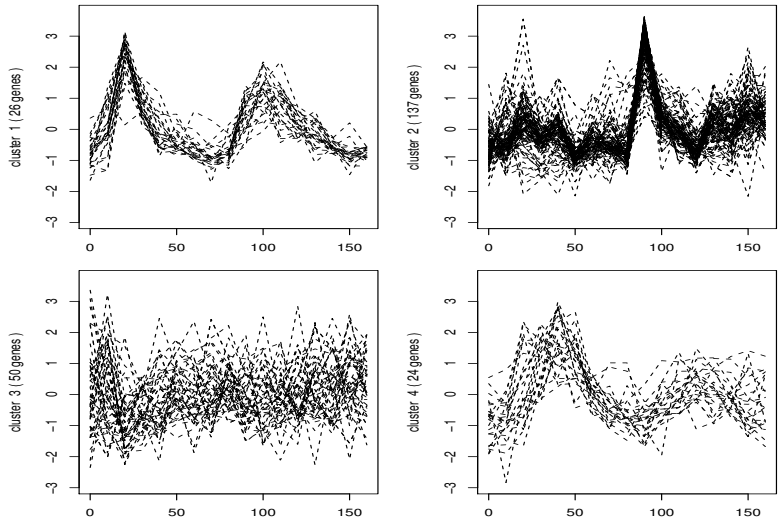


Figure 1: Clusters of Gene-Profiles Obtained by Mixture of Linear Mixed Models with Cluster-Specific Random Effects

The clustering obtained in the latter case, however, is still superior in terms of the Rand Index and its adjusted value for the two-stage method of Wong et al. [10], which was the best on the basis of these criteria in their comparative analysis. We also fitted the mixed linear model mixture (4) without

Table 2: Summary of Clustering Results for $g = 4$ Clusters

Model	Rand Index	Adjusted Rand Index	Error Rate
1	0.7808	0.5455	0.291
2	0.7152	0.4442	0.316
3	0.7133	0.3792	0.4093
Wong	0.7087	0.3697	Not available

the sine-cos regression model (5) for the mean, but with a separate (fixed effects) term at each of

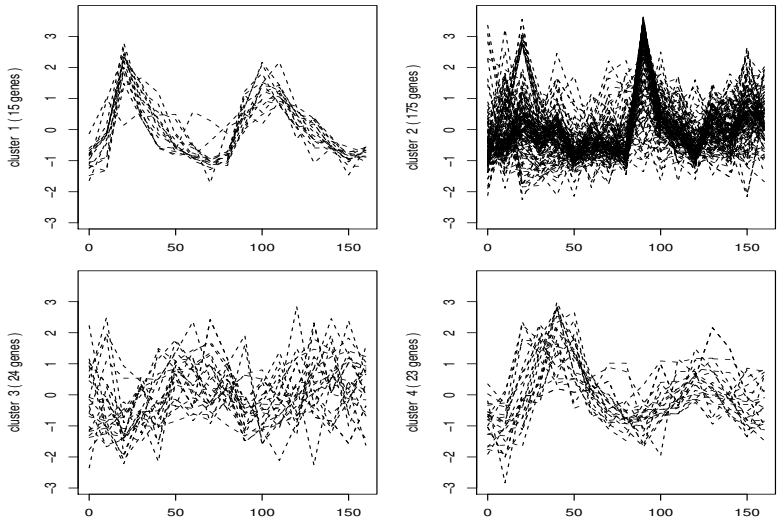


Figure 2: Clusters of Gene-Profiles Obtained by Mixture of Linear Mixed Models without Cluster-Specific Random Effects

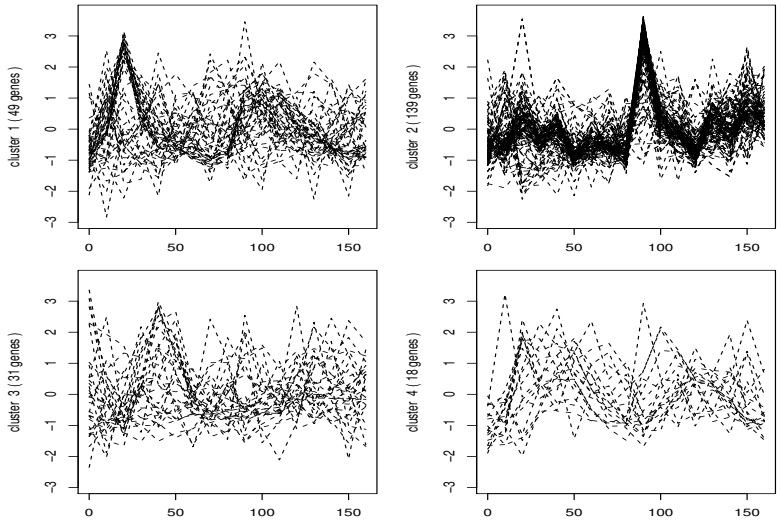


Figure 3: Plots of Gene Profiles Grouped According to Their True Functional Grouping

the $p = 17$ time points; that is, we set $\boldsymbol{X} = \boldsymbol{I}_p$ and took β_i to be a p-dimensional vector of fixed effects. We did not include cluster-specific random-effects terms $\boldsymbol{c}_i$ due to their nonidentifiability in this case. This nonregression model gave worse results for the Rand Index and the error rate than with the regression model (4) using the sine-cos curve to specify the mean at a given time point. The results for this nonregression version are listed under Model 3 in Table 2, where the clustering results have been summarized. In this table, Models 1 and 2 correspond to the use of the regression model (4) with and without cluster-specific random-effects terms.

In Figures 1 and 2, we give the plots of the gene profiles as clustered into $g = 4$ clusters as obtained by fitting the mixture of linear mixed models (4) with and without cluster-specific random-effects terms $\boldsymbol{c}_i$. In Figure 3, the plots of the gene profiles are grouped according to their actual functional grouping.

Acknowledgments

The authors would like to acknowledge the support of the Australian Research Council.

References

[1] McLACHLAN, G.J. & PEEL, D. (2000) *Finite Mixture Models*. Wiley, New York.

[2] NG, S.K., McLACHLAN, G.J., WANG, K., BEN-TOVIM JONES, L. & Ng, S.W. (2006) A mixture model with random-effects components for clustering correlated gene-expression profiles. *Bioinformatics* 22, pp. 1745-1752.

[3] McLACHLAN, G.J., DO, K.-A. & AMBROISE, C. (2004) *Analyzing Microarray Gene Expression Data*. Wiley, Hoboken, New Jersey.

[4] LUAN, Y. & LI, H. (2003) Clustering of time-course gene expression data using a mixed-effects model with B-splines. *Bioinformatics* 19, pp. 474-482.

[5] CELEUX, G., MARTIN, O. & LAVERGNE, C. (2005) Mixture of linear mixed models for clustering gene expression profiles from repeated microarray experiments. *Statistical Modelling* 5, pp. 243-267.

[6] QIN, L.-X. & SELF, S.G. (2006) The clustering of regression models method with applications in gene expression data. *Biometrics* 62, pp. 526-533.

[7] BOOTH, J.G., CASELLA, G. & HOBERT, J.P. (2008) Clustering using objective functions and stochastic search. *Journal of the Royal Statistical Society* B 70, pp. 119-139.

[8] CHO, R.J., HUANG, M., CAMPBELL, M.J. et al. (2001) Transcriptional regulation and function during the human cell cycle. *Nature Genetics* 27, pp. 48-54.

[9] YEUNG, K.Y. , FRALEY, C., MURUA, A., RAFTERY, A.E. & RUZZO, W.L. (2001) Model-based clustering and data transformations for gene expression data. *Bioinformatics* 17, pp. 977-987.

[10] WONG, D.S.V., WONG, F.K. & WOOD, G.R. (2007) A multi-stage approach to clustering and imputation of gene expression profiles. *Bioinformatics* 23, pp. 998-1005.

Stationary long-memory process in the presence of additive outliers. A robust model estimation.

Valdério A. Reisen
Universidade Federal do Espírito Santo, Brazil.
valderio@cce.ufes.br

Fabio A. Fajardo Molinares
Núcleo de Modelagem Estocástica - NuMes, UFES, Brazil.
ffajardo@est.dout.ufmg.br

Francisco Cribari-Neto
Universidade Federal de Pernambuco, Brazil.
cribari@ufpe.br

Abstract

In this paper, we introduce an alternative semiparametric estimator of the fractional differencing parameter in ARFIMA models which is robust against additive outliers. The proposed estimator is a variant of the GPH estimator (Geweke & Porter-Hudak(1983)). In particular, we use the robust sample autocorrelations of Ma & Genton (2000) to obtain an estimator for the spectral density of the process. Numerical results show that the estimator we propose for the differencing parameter is robust when the data contain additive outliers.

1 Introduction

In the early 1980s, Granger & Joyeux (1980) and Hosking (1981) proposed an extension of the ARIMA process in which the differecing parameter is allowed to assume non-integer values: the ARFIMA process. Hosking (1981) showed that series with ARFIMA representation for which $d \in (0, 0.5)$ are stationary and display long memory; the latter is expressed by statistically significante autocorrelations at large lags or, alternatively, by a singularity of the spectral density at the zero frequency.

Haldrup & Nielsen (2007) evaluated the impacts of measurement errors, outliers and structural breaks on the estimation of the long-memory parameter. The results show that such impacts can be quite substantial. For instance, an additive outlier in the data may substantially bias the differencing parameter estimate. They concluded that the regression-based semi-parametric estimators are less biased when the bandwidth, which corresponds to the number of frequencies used in the estimation, is small. The authors suggested the use of the approach proposed by Sun & Phillips (2003), which adds a nonlinear factor to the log-periodogram regression, as a way to minimize any existing bias. In a similar context, Agostinelli & Bisaglia (2003) proposed the use of a weighted maximum likelihood approach as a modification of the the estimator proposed by Beran (1994).

In this paper we propose a robust estimator for the long-memory parameter in ARFIMA processes. The proposed estimator is robust against additive outliers. The estimation approach we use is based on the robust estimator of the autocovariance function proposed by Ma & Genton (2000) in the context of obtaining the periodogram. The robust long-memory estimator is a variant of the well known method proposed by Geweke & Porter-Hudak (1983) (GPH). We show the robustness of our estimator both analytically and through Monte Carlo simulations.

In what follows, we shall consider the case where the process $\{y_t\}_{t\in\mathbb{Z}}$ is stationary ARFIMA(p, d, q).

2 Stationary ARFIMA process

Let $\{y_t\}_{t\in\mathbb{Z}}$ be a linear process such that

$$\Phi(B)y_t = \Theta(B)(1-B)^{-d}\epsilon_t, \quad \text{with } d \in (-0.5, 0.5),$$

where $\Phi(x) = 1 - \phi_1 x - \cdots - \phi_p x^p$ and $\Theta(x) = 1 - \theta_1 x - \cdots - \theta_p x^p$ are polynomials with no common roots and with all roots outside the unit circle; here, $\{\epsilon_t\}$ is a zero mean white noise process with variance σ_ϵ^2. That is, $\{y_t\}$ follows a stationary and invertible ARFIMA(p, d, q) process. Note that the fractional differencing filter $(1-B)^d$, for $d \in \mathbb{R}$, is defined by the binomial expansion

$$(1-B)^d = \sum_{j=0}^{\infty} \zeta_j B^j,$$

where $\zeta_j = \frac{\Gamma(j-d)}{\Gamma(j+1)\Gamma(-d)}$, $j = 0, 1, 2, \ldots$, $\Gamma(\cdot)$ being the Gamma function.

The spectral density function of $\{y_t\}$ is given by

$$f_y(\lambda) = \frac{\sigma_\epsilon^2}{2\pi} \frac{|\Theta(e^{-i\lambda})|^2}{|\Phi(e^{-i\lambda})|^2} \left\{2\sin\left(\frac{\lambda}{2}\right)\right\}^{-2d}; \tag{1}$$

see Hosking (1981) and Reisen (1994) for details.

3 Long-memory parameter estimators

Let $f_y(\lambda_j)$ be as in (1), for $\lambda_j = \frac{2\pi j}{n}$, $j = 0, 1, \ldots, \lfloor \frac{n}{2} \rfloor$, where n is the sample size and $\lfloor x \rfloor$ denotes the integer part of x. The natural logarithm of the spectral density $f_y(\lambda_j)$ is

$$\log f_y(\lambda_j) = \log f_u(0) - d \log\left\{2\sin\left(\frac{\lambda_j}{2}\right)\right\}^2 + \log\frac{f_u(\lambda_j)}{f_u(0)}, \tag{2}$$

where $f_u(\lambda)$ is the spectral density of $U_t = (1-B)^d y_t$.

Using (2), Geweke & Porter-Hudak (1983) proposed a semiparametric estimator of d. We shall use equation (2) to obtain a robust estimator for the long-memory parameter.

3.1 The GPH estimator

By adding $\log I(\lambda)$ to both sides of (2) and considering frequencies close to zero, an estimate of d can be obtained from the following regression equation:

$$\log I(\lambda_j) = \beta_0 - d \log\left\{2\sin\left(\frac{\lambda_j}{2}\right)\right\}^2 + e_j, \quad j = 1, 2, \ldots, g(n), \tag{3}$$

where $\beta_0 = \log f_u(0) + \log\frac{f_u(\lambda_j)}{f_u(0)} + c$, $e_j = \log\frac{I(\lambda_j)}{f(\lambda_j)} - c$ and $c = \psi(1)$, where $\psi(\cdot)$ is the digamma fuction, i.e. $\psi(a) = \frac{\partial \log\Gamma(a)}{\partial a}$.

Geweke & Porter-Hudak (1983) established that, for $d < 0$ and $\left\{\log\frac{I(\lambda_j)}{f(\lambda_j)}\right\}_{j=1,\ldots,g(n)}$ is a sequence of approximately independent Gumbel random variables with mean $\psi(1)$ and variance $\frac{\pi^2}{6}$. Hence, $\{e_j\}$ is a sequence of approximately independent Gumbel variables with mean zero and variance $\frac{\pi^2}{6}$.

The GPH estimator is given by

$$d_{GPH} = -\frac{\sum_{j=1}^{g(n)}(x_j - \bar{x})\log I(\lambda_j)}{\sum_{j=1}^{g(n)}(x_j - \bar{x})^2}, \tag{4}$$

where $x_j = \log\left\{2\sin\left(\frac{\lambda_j}{2}\right)\right\}^2$, $g(n)$ being the bandwidth in the regression equation which has to satisfy $g(n) \to \infty, n \to \infty$, with $\frac{g(n)}{n} \to 0$. The variance of the GPH estimator is

$$\mathrm{var}(d_{GPH}) = \frac{\pi^2}{6\sum_{j=1}^{g(n)}(x_j - \bar{x})^2}.$$

Geweke & Porter-Hudak (1983) proved the asymptotic normality of the semiparametric estimator in (4) when $d < 0$ and suggested taking $g(n) = n^\alpha$, $0 < \alpha < 1$. However, the asymptotic properties established originally for the GPH estimator were contested by Künsch (1986), Hurvich & Beltrão (1993) and Robinson (1995) for $d \neq 0$. Hurvich, Deo & Brodsky (1998) proved that, under some regularity conditions on the choice of the bandwidth, the GPH estimator is consistent for the memory parameter and is asymptotically normal when the time series is Gaussian. The authors also established that the optimal $g(n)$ in equation (3) is of order $o(n^{4/5})$. They showed that if $g(n) \to \infty, n \to \infty$ with $\frac{g(n)}{n} \to 0$ and $\frac{g(n)}{n}\log g(n) \to 0$, then, under some conditions on $0 < f_u(\lambda_j) < \infty$, the GPH estimator is a consistent estimator of $d \in (-0.5, 0.5)$ with variance $\mathrm{var}(d_{GPH}) = \frac{\pi^2}{24g(n)} + o(g(n)^{-1})$. If $g(n) = o(n^{4/5})$ and $\log^2 n = o(g(n))$, then

$$\sqrt{g(n)}(d_{GPH} - d) \rightsquigarrow N\left(0, \frac{\pi^2}{24}\right),$$

where $\rightsquigarrow$ denotes convergence in distribution.

4 Robust estimation

We shall use the robust correlogram due to Ma & Genton (2000) to obtain a robust estimator for the spectral density function and, as a consequence, a robust estimator for the long-memory parameter d (say, d_{GPHR}).

4.1 Robust estimators for the autocovariance and spectral density functions

Ma & Genton (2000) proposed a robust estimator for the ACOVF based on a scale approximation for the covariance between two random variables and on the estimator $Q_n(\cdot)$, proposed by Rousseeuw & Croux (1993). The estimator $Q_n(\cdot)$ is based on the kth order statistic of the $\binom{n}{2}$ distances $\{|z_i - z_j|, i < j\}$, and can be written as

$$Q_n(z) = c \times \{|z_i - z_j|; i < j\}_{(k)}, \tag{5}$$

where $z = (z_1, z_2, \ldots, z_n)'$, c is a constant used to guarantee consistency ($c = 2.2191$ for the normal distribution), and $k = \left\lfloor \frac{\binom{n}{2}+2}{4} \right\rfloor + 1$. One can use the algorithm proposed by Croux & Rousseeuw (1992), which is computationally efficient. The robust estimator for the ACOVF can be expressed as

$$\widetilde{R}(h) = \frac{1}{4}\left[Q^2_{n-h}(u+v) - Q^2_{n-h}(u-v)\right], \tag{6}$$

where u and v are vectors contains the initial $n-h$ and the last $n-h$ observations, respectively. The robust estimator for the autocorrelation function is

$$\widetilde{\rho}(h) = \frac{Q^2_{n-h}(u+v) - Q^2_{n-h}(u-v)}{Q^2_{n-h}(u+v) + Q^2_{n-h}(u-v)},$$

which is such that $|\widetilde{\rho}(h)| \leq 1$.

Ma & Genton (2000) proved the robustness of (6) and showed that its variance cannot be written in closed form.

4.2 A robust estimator of d

As described below, the robust correlogram, introduced in Section 4.1, can be used to obtain a robust periodogram.

Let $\widetilde{I}(\lambda)$ be given by

$$\widetilde{I}(\lambda) = \frac{1}{2\pi} \sum_{s=-(n-1)}^{n-1} \kappa(s)\widetilde{R}(s)\cos(s\lambda), \tag{7}$$

where $\widetilde{R}(s)$ is the sample autocovariance function in (6) and $\kappa(s)$ is defined as

$$\kappa(s) = \begin{cases} 1, & |s| \leq M, \\ 0, & |s| > M, \end{cases}$$

along with $M = n^{\beta}$, $0 < \beta < 1$. $\kappa(s)$ is called *truncated periodogram lag window* see, e.g., Priestley (1981, pp. 433-437). We shall call the estimator in (7) *robust truncated pseudo-periodogram*, since it does not have the same finite-sample properties as the periodogram.

Based on the above and using the regression equation given in (3), the robust GPH estimator we propose is

$$d_{GPHR} = -\frac{\sum_{i=1}^{g(n)}(x_i - \bar{x})\log\widetilde{I}(\lambda_i)}{\sum_{i=1}^{g(n)}(x_i - \bar{x})^2}, \tag{8}$$

where $x_i = \log\left\{2\sin\left(\frac{\lambda_j}{2}\right)\right\}^2$ and $g(n)$ is as before.

5 Empirical evidences

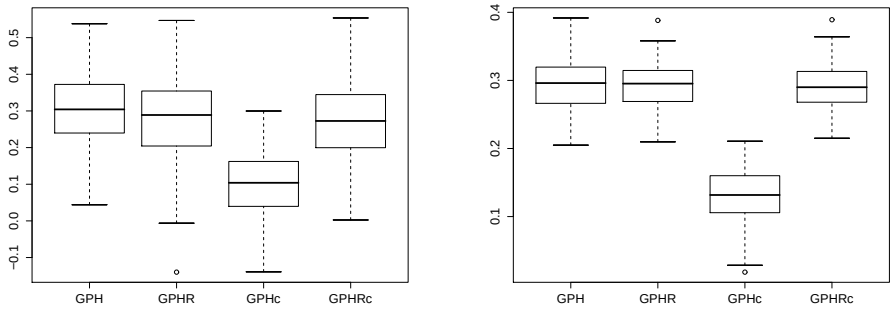
In order to investigate the empirical properties of the proposed estimator, a number of Monte Carlo experiments were carried out.

Realizations of a Gaussian white noise sequence $\{\varepsilon_t\}_{t=1,\cdots,n}$, with zero mean and variance 1, were generated by the function `rann` of the `Ox` matrix programming language. The long-memory time series with size n was simulated according to Hosking (1981) with 5% of outliers of magnitude 10. The bandwidth $g(n)$ was fixed at $n^{0.7}$ (see Reisen (1994)). Figure 1 gives the box-plots of the estimates when d =0.3 and n = 300, 3000. GPH and GPHR mean, respectively, the non-robust and robust estimates of the memory parameter in a time series without outliers while GPHc and GPHRc are the estimates when the series is contaminated by inconsistent observations. The boxplots clearly evidence that the proposed method is robust in the presence of outliers. The figures also show that the GPH estimates underestimate significantly the true parameter when the series has outliers. The methods have similar performance for uncontaminated data. The increasing of the sample size reduces significantly the bias and variance of the robust estimation method.

Table 1 presents results for $d = 0.3, 0.45$ and $\alpha = \beta = 0.7$; the columns d_{GPHc} and d_{GPHRc} contain results for the outlier contaminated series. The figures in Table 1 show that the GPH estimator is sensitive to outliers; in particular, it underestimates the true parameter value when the data contain atypical observations. It is noteworthy that the finite-sample behavior of the robust estimator proposed in this paper is not affected by the introduction of atypical observations in the data; its performance improves rapidly with the sample size (see Figure 1).

Table 2 shows that the bias of the estimator d_{GPHc} depends on ω. The bias of d_{GPHRc}, on the other hand, is insensitive to ω.

The paper also investigate the order identification and parameter estimation of full ARFIMA models. The procedure is described below. The empirical results are not presented here but available with the authors and they indicated that the procedure is very promising in estimating the parameters of the ARFIMA processes in the presence of outliers and can also easily be used in real situations.



(a) Sample size $n = 300$ (b) Sample size $n = 3000$

Figure 1: Estimates of parameter d for n=300, 3000 obtained for the estimators for contaminated data (GPHc and GPHRc, respectively) and outlier-free data (GPH and GPHR).

d	n		d_{GPH}	d_{GPH_c}	d_{GPHR}	d_{GPHR_c}
0.30	100	mean	0.2988	0.1134	0.2584	0.2449
		s.d.	0.1735	0.1619	0.1558	0.1556
		bias	−0.0012	−0.1866	−0.0416	−0.0551
		MSE	0.0301	0.0610	0.0260	0.0272
	300	mean	0.3062	0.1007	0.2907	0.2837
		s.d.	0.1005	0.0978	0.0926	0.0960
		bias	0.0062	−0.1993	−0.0093	−0.0163
		MSE	0.0101	0.0493	0.0087	0.0095
	800	mean	0.3003	0.1184	0.2949	0.2869
		s.d.	0.0679	0.0715	0.0573	0.0610
		bias	0.0003	−0.1816	−0.0051	−0.0131
		MSE	0.0046	0.0381	0.0033	0.0039
0.45	100	mean	0.4561	0.1923	0.3975	0.3778
		s.d.	0.1722	0.1727	0.1506	0.1433
		bias	0.0061	−0.2577	−0.0525	−0.0722
		MSE	0.0297	0.0962	0.0254	0.0258
	300	mean	0.4594	0.2015	0.4329	0.4233
		s.d.	0.0986	0.0976	0.1041	0.1013
		bias	0.0094	−0.2485	−0.0171	−0.0267
		MSE	0.0098	0.0713	0.0111	0.0110
	800	mean	0.4620	0.2306	0.4457	0.4349
		s.d.	0.0688	0.0809	0.0562	0.0576
		bias	0.0121	−0.2194	−0.0043	−0.0151
		MSE	0.0049	0.0547	0.0032	0.0035

Table 1: Simulation results; ARFIMA$(0, d, 0)$ with $\alpha = \beta = 0.7$ and $\omega = 0, 10$.

ω	n		d_{GPH_c}	d_{GPHR_c}
3	100	mean	0.3747	0.3799
		s.d.	0.1953	0.1513
		bias	−0.0753	−0.0701
		MSE	0.0438	0.0278
	800	mean	0.4080	0.4309
		s.d.	0.0679	0.0576
		bias	−0.0419	−0.0191
		MSE	0.0064	0.0037
5	100	mean	0.3108	0.3741
		s.d.	0.1934	0.1452
		bias	−0.1392	−0.0759
		MSE	0.0567	0.0268
	800	mean	0.3526	0.4270
		s.d.	0.0846	0.0568
		bias	−0.0974	−0.0229
		MSE	0.0166	0.0038
10	100	mean	0.1923	0.3778
		s.d.	0.1727	0.1433
		bias	−0.2577	−0.0722
		MSE	0.0962	0.0258
	800	mean	0.2306	0.4349
		s.d.	0.0809	0.0576
		bias	−0.2194	−0.0151
		MSE	0.0547	0.0035

Table 2: Simulation results; ARFIMA$(0, d, 0)$ with $d = 0.45$, $\omega = 3, 5, 10$ and $\alpha = \beta = 0.7$.

5.1 ARFIMA parameter estimation

1. Estimate d using the robust GPH estimator, $\hat{d} = d_{GPHR}$.
2. Compute $\widehat{U}_t = (1 - B)^{\hat{d}} z_t$.
3. For $\Phi(B)\widehat{U}_t = \Theta(B)\epsilon_t$, use the Box-Jenkins approach to identify the orders of the autoregressive and moving average polynomials, and then estimate $\phi_1, \phi_2, \ldots, \phi_p$ and $\theta_1, \theta_2, \ldots, \theta_q$.
   Note: In this step we use $\widetilde{\rho}(\lambda)$ in the Yule-Walker equations to obtain the estimates of ARMA components.
4. Perform the usual goodness-of-fit tests.

6 Applications

We have applied the methodology proposed in the annual minimum water levels of the Nile river measured at the Roda gorge (see, e.g. Hosking (1981), Beran (1994). The period analyzed for this data set was from 622 A.D. to 1284 A.D. and displayed in Figure 2. The estimated values for parameter d, and other proposed alternatives, are presented in Table 3.

	$\widehat{d}$
Robinson (1994)	0.4338
Beran (1994)	0.4000
Agostinelli & Bisaglia (2003)	0.4160
GPH Robusto (d_{GPHR})	0.4161

Table 3: Estimated values of the parameter d: *Annual minimum levels of the Nile river.*

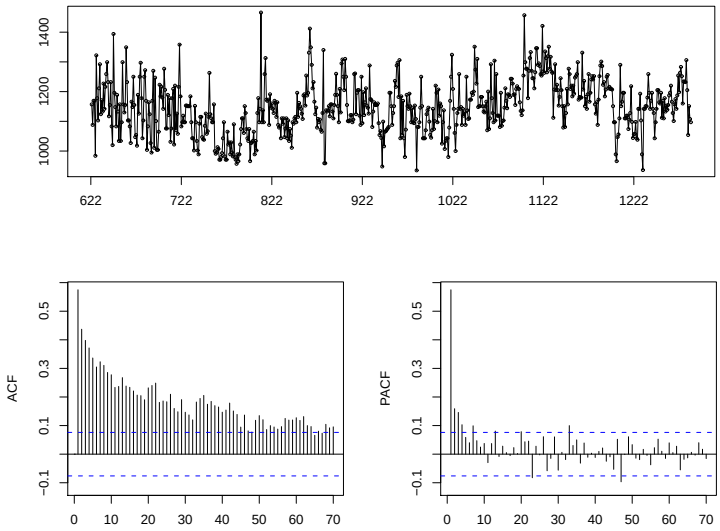


Figure 2: Annual minimum levels of the Nile river with sample ACF and PACF.

Acknowledgments

The financial support from CAPES and CNPq is gratefully acknowledged.

References

[1] Agostinelli, C. & Bisaglia, L. (2003). Robust estimation of ARFIMA processes. *Technical Report*, Università Cà Foscari di Venezia.

[2] Beran, J. (1994). On class of M-estimators for Gaussian long-memory models. *Biometrika* **81**, pp. 755-766.

[3] Croux, C. & Rousseeuw, P. (1992). Time-efficient algorithms for two highly robust estimators of scale. *Computational Statistics* **1**, pp. 1-18.

[4] Geweke, J. & Porter-Hudak, S. (1983). The estimation and application of long memory time series model. *Journal of Time Series Analysis* **4**, pp. 221-238.

[5] Granger, C. W. & Joyeux, R. (1980). An introduction to long-memory time series models and fractional differencing. *Journal of Time Series Analysis* **1**, pp. 15-30.

[6] Haldrup, N. & Nielsen, M. (2007). Estimation of fractional integration in the presence of data noise. *Computational Statistical & Data Analysis* **51**, pp. 3100-3114.

[7] Hosking, J. R. (1981). Fractional differencing. *Biometrika* **68**, pp. 165-176.

[8] Hurvich, C. M., Deo R. & Brodsky, J. (1998). The mean square error of Geweke and Porter-Hudak's estimator of the memory parameter of a long-memory time series . *Journal of Time Series Analysis* **19**, pp. 19-46.

[9] Künsch, H. R. (1986). Discrimination between monotonic trends and log-range dependence. *Journal of Applied Probability* **23**, pp. 1025-1030.

[10] Ma, Y. & Genton, M. (2000). Highly robust estimation of the autocovariance function. *Journal of Time Series Analysis* **21**, pp. 663-684.

[11] Reisen, V. (1994). Estimation of the fractional difference parameter in the ARIMA(p, d, q) model using the smoothed periodogram *Journal of Time Series Analysis* **15**, pp. 335-350.

[12] Robinson, P. (1994). Semiparametric analysis of long-memory time series *The Annals of Statistics* **22**, pp. 515-539.

[13] Robinson, P. (1995). Lo-periodogram regression of time series with long range dependence *The Annals of Statistics* **23**, pp. 1048-1072.

[14] Rousseeuw, P. & Croux, C. (1993). Alternatives to the median absolute deviation. *Journal of the American Statistical Association* **88**, pp. 1273-1283.

[15] Sun, Y. & Phillips, P. (2003). Nonlinear log-periodogram regression for perturbed fractional process. *Journal of Econometrics* **115**, pp. 355-389.

Parameter Estimation Procedures in Time Series Models

Teo Sharia
Department of Mathematics
Royal Holloway, University of London
Egham, Surrey TW20 0EX
t.sharia@rhul.ac.uk

Abstract

A wide class of asymptotically efficient parameter estimation procedures is proposed for general time series models. The procedures allow one to incorporate auxiliary information into the estimation process and under certain regularity conditions are consistent and asymptotically efficient.

1 Introduction

Consider an AR(1) process

$$X_t = \theta X_{t-1} + \xi_t, \tag{1}$$

where ξ_t is a sequence of random variables (r.v.'s) with mean zero. The least squares (LS) estimator $\hat{\theta}_t^{LS}$ of θ can be written recursively as

$$\hat{\theta}_t^{LS} = \hat{\theta}_{t-1}^{LS} + \hat{I}_t^{-1} X_{t-1} \left(X_t - \hat{\theta}_{t-1}^{LS} X_{t-1} \right), \tag{2}$$

$$\hat{I}_t = \hat{I}_{t-1} + X_{t-1}^2. \tag{3}$$

where $\hat{\theta}_0 = 0$ and $\hat{I}_0 = 0$. This can easily be verified by subtracting two successive terms of $\hat{\theta}_t^{LS} = \sum_{s=1}^t X_s X_{s-1} / \sum_{s=1}^t X_{s-1}^2$ and simple algebra (note also that $\hat{I}_t = \sum_{s=1}^t X_{s-1}^2$). It is well-known that in the case when ξ_t is a sequence of Gaussian i.i.d. r.v.'s, the LS estimators are consistent and asymptotically efficient. However, in the case of non-Gaussian ξ_t's the LS estimators fail to be efficient.

Suppose now that ξ_t is a sequence of i.i.d. r.v.'s and the probability density function of ξ_t w.r.t. Lebesgue's measure is $g(x)$. Consider an estimator defined recursively as

$$\hat{\theta}_t = \hat{\theta}_{t-1} - \hat{I}_t^{-1} i^{g-1} X_{t-1} \frac{g'(X_t - \hat{\theta}_{t-1} X_{t-1})}{g(X_t - \hat{\theta}_{t-1} X_{t-1})}, \tag{4}$$

where $i^g = \int \left(g'(z)/g(z)\right)^2 g(z)\, dz, \quad t \geq 1$ and $\hat{\theta}_0 \in \mathbb{R}$ is an arbitrary starting point. We will refer to (4) as the recursive likelihood procedure. This name can be justified by the fact that under certain conditions, the estimators defined by (4) are asymptotically equivalent to MLEs in the sense that they have the same asymptotic properties as the MLE's, in particular consistency and asymptotic efficiency. A heuristic justification of the estimation procedures of this type in a more general setting will be given later in the paper.

Let us now consider a class of estimation procedures defined by

$$\hat{\theta}_t = \left[\hat{\theta}_{t-1} + \Gamma_t^{-1} \gamma(X_{t-1}) \phi(X_t - \hat{\theta}_{t-1} X_{t-1}) \right]_{\alpha_t}^{\beta_t}, \tag{5}$$

with suitably chosen ϕ, γ, and Γ_t. Here α_t and β_t are random variables with $-\infty \leq \alpha_t \leq \beta_t \leq \infty$ and $[v]_{\alpha_t}^{\beta_t}$ is the truncation operator, that is,

$$[v]_{\alpha_t}^{\beta_t} = \begin{cases} \alpha_t & \text{if } v < \alpha_t \\ v & \text{if } \alpha_t \leq v \leq \beta_t \\ \beta_t & \text{if } v > \beta_t. \end{cases}$$

The truncation interval $U_t = [\alpha_t, \beta_t]$ represents our auxiliary knowledge about the unknown parameter which is incorporated in the procedure through the truncation operator. For example, if $\theta \in \Theta = [\alpha, \beta]$, then one can take $\alpha_t = \alpha$ and $\beta_t = \beta$. In the case of the open interval $\Theta = (\alpha, \beta)$ we may choose to consider truncations with moving bounds to avoid possible singularities at the endpoints of the interval. That is, we can take $U_t = [\alpha_t, \beta_t]$ with some sequences $\alpha_t \downarrow \alpha$ and $\beta_t \uparrow \beta$.

The most interesting case arises when a consistent, but not necessarily efficient auxiliary estimator $\tilde{\theta}_t$ is available having a rate d_t. Then one can use $\tilde{\theta}_t$ to truncate the recursive procedure in a neighbourhood of θ by taking $U_t = [\tilde{\theta}_t - \varepsilon_t, \tilde{\theta}_t + \varepsilon_t]$ with $\varepsilon_t \to 0$. Such a procedure is obviously consistent since $\hat{\theta}_t \in [\tilde{\theta}_t - \varepsilon_t, \tilde{\theta}_t + \varepsilon_t]$ and $\tilde{\theta}_t \pm \varepsilon_t \to \theta$. However, since our main goal is to construct an efficient estimator, care should be taken to ensure that the truncation intervals do not shrink to θ too rapidly, for otherwise $\hat{\theta}_t$ will have the same asymptotic properties as $\tilde{\theta}_t$.

An example of possible applications of (5) is a likelihood procedure with LS truncations, that is,

$$\hat{\theta}_t = \left[\hat{\theta}_{t-1} - \hat{I}_t^{-1} i_g^{-1} X_{t-1} \frac{g'(X_t - \hat{\theta}_{t-1} X_{t-1})}{g(X_t - \hat{\theta}_{t-1} X_{t-1})} \right]_{\hat{\theta}_t^{LS} - c\hat{I}_t^{-\varepsilon}}^{\hat{\theta}_t^{LS} + c\hat{I}_t^{-\varepsilon}} \tag{6}$$

where $\hat{\theta}_t^{LS}$ and $\hat{I}_t$ are defined by (2) and (3), and c and ε are positive constants.

Let us now consider a general time series model given by a sequence $X_1, \ldots, X_t$ of r.v.'s with the joint distribution depending on an unknown parameter $\theta \in \mathbb{R}^m$. Recall that an M-estimator of θ is defined as a solution of the estimating equation

$$\sum_{s=1}^{t} \psi_s(v) = 0, \tag{7}$$

where $\psi_s(v) = \psi_s(X_1^s; v)$, $s = 1, 2, \ldots, t$, are suitably chosen functions which may, in general, depend on the vector $X_1^s = (X_1, \ldots, X_s)$ of all past and present observations. If $f_s(x, \theta) = f_s(x, \theta | X_1, \ldots, X_{s-1})$ is the conditional probability density function (pdf) or probability function (pf) of the observation X_s given $X_1, \ldots, X_{s-1}$, then one can obtain a MLE on choosing $\psi_s(v) = f_s'(X_s, v)/f_s(X_s, v)$. Besides MLEs, the class of M-estimators includes estimators with special properties such as robustness. Under certain regularity and ergodicity conditions, there exists a consistent sequence of solutions of (7) which has the property of local asymptotic linearity.

If ψ-functions are nonlinear, it is rather difficult to work with the corresponding estimating equations. Note that for a linear estimator, e.g., for the sample mean $\hat{\theta}_t = \bar{X}_t$, we have $\bar{X}_t = (t-1)\bar{X}_{t-1}/t + X_t/t$, that is $\hat{\theta}_t = \hat{\theta}_{t-1}(t-1)/t + X_t/t$, which means that the estimator $\hat{\theta}_t$ at each step t can be obtained recursively using the estimator at the previous step $\hat{\theta}_{t-1}$ and the new information X_t. Such an exact recursive relation may not hold for nonlinear estimators (e.g., in the case of the median).

In general, to find a possible form of an approximate recursive relation consider $\hat{\theta}_t$ defined as a root of the estimating equation (7). Denoting the left hand side of (7) by $M_t(v)$ and assuming that the difference $\hat{\theta}_t - \hat{\theta}_{t-1}$ is "small" we can write $M_t(\hat{\theta}_t) \approx M_t(\hat{\theta}_{t-1}) + M_t'(\hat{\theta}_{t-1})(\hat{\theta}_t - \hat{\theta}_{t-1})$ and

$$0 = M_t(\hat{\theta}_t) - M_{t-1}(\hat{\theta}_{t-1}) \approx M_t'(\hat{\theta}_{t-1})(\hat{\theta}_t - \hat{\theta}_{t-1}) + \psi_t(\hat{\theta}_{t-1}).$$

Therefore,

$$\hat{\theta}_t \approx \hat{\theta}_{t-1} - \frac{\psi_t(\hat{\theta}_{t-1})}{M_t'(\hat{\theta}_{t-1})},$$

where $M_t'(\theta) = \sum_{s=1}^{t} \psi_s'(\theta)$. Now, depending on the nature of the underlying model, $M_t'(\theta)$ can be replaced by a simpler expression. For instance, in the i.i.d. models with $\psi(x, v) = f'(x, v)/f(x, v)$ (the MLE case), by the strong law of large numbers,

$$\frac{M_t'(\theta)}{t} = \frac{1}{t} \sum_{s=1}^{t} \left(f'(X_s, \theta)/f(X_s, \theta) \right)' \approx E_\theta \left[\left(f'(X_1, \theta)/f(X_1, \theta) \right)' \right] = -i(\theta)$$

for large t's, where $i(\theta)$ is the one-step Fisher information. So, in this case, one can consider

$$\hat{\theta}_t = \hat{\theta}_{t-1} + \frac{1}{t\, i(\hat{\theta}_{t-1})} \frac{f'(X_t, \hat{\theta}_{t-1})}{f(X_t, \hat{\theta}_{t-1})}, \qquad t \geq 1, \tag{8}$$

to construct an estimator which is "asymptotically equivalent" to the MLE.

Motivated by the above argument, we consider a class of estimators

$$\hat{\theta}_t = \hat{\theta}_{t-1} + \Gamma_t^{-1}(\hat{\theta}_{t-1})\psi_t(\hat{\theta}_{t-1}), \qquad t \geq 1, \tag{9}$$

where ψ_t is a suitably chosen vector process, Γ_t is a (possibly random) normalizing matrix process, $\hat{\theta}_0 \in \mathbb{R}^m$ is some initial value. In particular, if $\psi_s(\theta) = f_s'(X_s, \theta)/f_s(X_s, \theta)$, where $f_s(x, \theta) = f_s(x, \theta | X_1, \ldots, X_{s-1})$ is the conditional pdf/pf of the observation X_s given $X_1, \ldots, X_{s-1}$, we obtain

$$\hat{\theta}_t = \hat{\theta}_{t-1} + I_t^{-1}(\hat{\theta}_{t-1}) \frac{f_t'^T(X_t, \hat{\theta}_{t-1})}{f_t(X_t, \hat{\theta}_{t-1})}, \qquad t \geq 1, \tag{10}$$

where, $I_t(\theta)$ is the conditional Fisher information matrix, f_t' is the row-vector of partial derivatives of f_t w.r.t. the components of θ (here T means transposition).

Now, it is easy to see that (4) is of the form of (10), since in this case, $f_s(x, \theta) = f_s(x, \theta | X_{s-1}) = g(X_t - \theta X_{t-1})$ and $I_t(\theta) = i_g \hat{I}_t = i_g \sum_{i=1}^t X_{s-1}^2$.

It should be noted that at first glance, recursions (8) and (10) resemble the Newton-Raphson or the one-step Newton-Raphson iterative procedure of numerical optimisation. In the i.i.d. case, the Newton-Raphson iteration for the likelihood equation is

$$\vartheta_k = \vartheta_{k-1} + J^{-1}(\vartheta_{k-1}) \sum_{s=1}^t \frac{f'(X_s, \vartheta_{k-1})}{f(X_s, \vartheta_{k-1})}, \qquad k \geq 1, \tag{11}$$

where $J(v)$ is minus the second derivative of the log-likelihood function, that is, $-\sum_{s=1}^t \frac{\partial}{\partial v}\left(f'(X_s, v)/f(X_s, v)\right)$ or its expectation, that is, the information matrix $ti(v)$. In the latter case, the iterative scheme is often called the method of scoring. The main feature of the scheme (11) is that t is fixed, and ϑ_k, at each step $k = 1, 2, \ldots$, is the k'th approximation to a root, say $\tilde{\theta}_t$, of the likelihood equation $\sum_{s=1}^t (f'(X_s, v)/f(X_s, v)) = 0$. Also, if a new $(t+1)$st observation is available, the whole procedure has to be repeated again. Note also, that the one-step Newton-Raphson is a simplified version of (11) when an auxiliary $\sqrt{t}$-consistent estimator, say $\tilde{\theta}_t$ is available. Then, the one-step Newton-Raphson improves $\tilde{\theta}_t$ in one step (that is, $k = 1$) by

$$\hat{\theta}_t = \tilde{\theta}_t + J^{-1}(\tilde{\theta}_t) \sum_{s=1}^t \frac{f'(X_s, \tilde{\theta}_t)}{f(X_s, \tilde{\theta}_t)}. \tag{12}$$

As we can see the procedure (8) is quite different. It does not require an auxiliary estimator and it adjusts the the value of the estimator in one single step at each instant of time with the arrival of the new observation. A theoretical implication of this is that by studying the procedures (8), or in general (9), we study the asymptotic behaviour of the estimator. As far as applications are concerned, there are several advantages in using (8), (9), or (10). Firstly, these procedures are easy to use and do not require storing all the data unnecessarily. This is especially convenient when the data come sequentially. Another potential benefit of using (9) is that it allows one to monitor and detect certain changes in probabilistic characteristics of the underlying process such as change of the value of the unknown parameter. So, there may be a benefit in using these procedures in linear cases as well.

Note also that the recursive procedure (9) is not a numerical solution of (7). Nevertheless, recursive estimator (9) and the corresponding M-estimator are expected to have the same asymptotic properties under quite mild conditions.

To understand how the procedure works, consider the likelihood recursive procedure (10) in the one-dimensional case. Denote $\Delta_t = \hat{\theta}_t - \theta$, rewrite the above recursion as

$$\Delta_t = \Delta_{t-1} + I_t^{-1}(\theta + \Delta_{t-1}) \frac{f_t'(X_t, \theta + \Delta_{t-1})}{f_t(X_t, \theta + \Delta_{t-1})}$$

and let

$$b_t(\theta, u) = E_\theta \left\{ \frac{f_t'(X_t, \theta + u)}{f_t(X_t, \theta + u)} \mid \mathcal{F}_{t-1} \right\},$$

where $\mathcal{F}_t$ is the σ-field generated by the random variables $X_1, \ldots, X_t$. Then,

$$E_\theta \left\{ \hat{\theta}_t - \hat{\theta}_{t-1} \mid \mathcal{F}_{t-1} \right\} = E_\theta \left\{ \Delta_t - \Delta_{t-1} \mid \mathcal{F}_{t-1} \right\} = I_t^{-1}(\theta + \Delta_{t-1}) b_t(\theta, \Delta_{t-1}).$$

Under usual regularity conditions (see [3] Remark 3.2 for details), $b_t(\theta, 0) = 0$ and $\frac{\partial}{\partial u} b_t(\theta, u) \mid_{u=0} = -i_t(\theta) < 0$, implying that

$$u b_t(\theta, u) < 0 \tag{13}$$

for small values of $u \neq 0$. Now, assuming that (13) holds for all $u \neq 0$, suppose that at time $t-1$, $\hat{\theta}_{t-1} < \theta$, that is $\Delta_{t-1} < 0$. Then, by (13), $E_\theta \left\{ \hat{\theta}_t - \hat{\theta}_{t-1} \mid \mathcal{F}_{t-1} \right\} > 0$. So, the next step $\hat{\theta}_t$ will be in the direction of θ. If at time $t-1$, $\hat{\theta}_{t-1} > \theta$, then by the same reason, $E_\theta \left\{ \hat{\theta}_t - \hat{\theta}_{t-1} \mid \mathcal{F}_{t-1} \right\} < 0$. So, on average, at each step the procedure moves towards θ. However, the magnitude of the jumps $\hat{\theta}_t - \hat{\theta}_{t-1}$ should decrease, for otherwise, $\hat{\theta}_t$ may oscillate around θ without approaching it. On the other hand, care should be taken to ensure that the jumps do not decrease too rapidly to avoid failure of $\hat{\theta}_t$ to reach θ.

Note also that in the iid case, (8) can be regarded as a stochastic iterative scheme, i.e., a classical stochastic approximation procedure, to detect the root of an unknown function when the latter can only be observed with random errors. To see this, let us rewrite (8) in terms of $\Delta_t = \hat{\theta}_t - \theta$ as

$$\Delta_t = \Delta_{t-1} + \frac{1}{ti(\theta + \Delta_{t-1})} \frac{f'(X_t, \theta + \Delta_{t-1})}{f(X_t, \theta + \Delta_{t-1})}.$$

Now, denoting

$$R^\theta(u) := \frac{1}{i(\theta + u)} E_\theta \left\{ \frac{f'}{f}(X_t, \theta + u) \right\}$$

we obtain

$$\Delta_t = \Delta_{t-1} + \frac{1}{t} \left(R^\theta(\theta + \Delta_{t-1}) + \varepsilon_t^\theta \right) \tag{14}$$

where

$$\varepsilon_t^\theta = \frac{1}{i(\theta + \Delta_{t-1})} \frac{f'}{f}(X_t, \theta + \Delta_{t-1}) - R^\theta(\Delta_{t-1}).$$

Under usual regularity conditions, $R(\theta, 0) = 0$ and $E_\theta(\varepsilon_t^\theta) = 0$. Equation (14) defines a Robbins-Monro stochastic approximation procedure that converges to the solution of the equation $R^\theta(u) = 0$ when the values of the function $R^\theta(u)$ can only be observed with zero expectation errors ε_t^θ. The technique of stochastic approximation has been exploited by a number of authors to study asymptotic behaviour of the recursive estimators in the iid case (see, e.g., [1] and [2] and references therein). Note that the idea of using auxiliary estimators in these schemes also goes back to [1] and [2]. Although in general, recursion (9) and (10) cannot be considered in the framework of classical stochastic approximation theory, some work has been done for non i.i.d. models as well. Discussion of these results and the references can be found in [3], [4] and [5].

2 Estimation using auxiliary information

Let us now return to a general time series model given by a sequence $X_1, \ldots, X_t$ of r.v.'s with the joint distribution depending on an unknown parameter $\theta \in \mathbb{R}^m$. It often happens that a statistician has auxiliary information which indicates in what range θ is likely to be:

- We may know a priori that the parameter lies in some set Θ. In this case it does not seem reasonable to use the procedure (9), especially as the functions in (9) may not be defined outside Θ. In this case one would want to have a procedure which generates values only from the set Θ.

- We may have an auxiliary estimator $\tilde{\theta}_t$ such that $d_t|\tilde{\theta}_t - \theta| \to 0$ as $t \to \infty$ (a.s.), where d_t is a sequence of positive numbers $d_t \uparrow \infty$.
- An interesting case arises when in estimating a multi-dimensional parameter, a qualitatively different additional (auxiliary) information is available for different components of θ, e.g., suppose $\theta = (\theta^{(1)}, \theta^{(2)})^T$ and we have an auxiliary consistent estimator only for the component $\theta^{(1)}$.

For a set $U \subseteq \mathbb{R}^m$, define a truncation operator as a function $\Phi_U : \mathbb{R}^m \longrightarrow \mathbb{R}^m$, such that

$$\begin{cases} \Phi_U(v) = v & \text{if } v \in U \\ \Phi_U(v) \in \text{closure}(U) & \text{if } v \notin U. \end{cases}$$

Suppose now that for each t a set U_t is given (which may depend on $X_1, \ldots, X_t$) such that $\theta \in U_t$ for large t's (a.s.). Define the recursive procedure by

$$\theta_t = \Phi_{U_t}\left(\theta_{t-1} + \Gamma_t^{-1}(\theta_{t-1})\psi_t(\theta_{t-1})\right). \tag{15}$$

In fact, U_t represents auxiliary knowledge about the unknown parameter which is incorporated in the procedure through the truncation operator Φ. For example, in the case (i) discussed above, if, e.g., $\theta \in \Theta$, then one can take $U_t = \Theta$ and

$$\Phi_{U_t}(v) = \begin{cases} v & \text{if } v \in \Theta \\ v* & \text{otherwise,} \end{cases}$$

where $v*$ denotes a closest point to v in the closure of the set Θ.

In the case when a consistent but not necessarily efficient auxiliary estimator $\tilde{\theta}_t$ is available having a rate d_t, a possible choice is $U_t = S(\tilde{\theta}_t, \gamma_t)$, where S is the ball in $\mathbb{R}^m$ with the center at $\tilde{\theta}_t$ and the radius $\delta_t = d_t^{-1} + \|\Gamma^{-1}(\tilde{\theta}_t)\|^{\varepsilon}$ ($\varepsilon < 1/2$), and

$$\Phi_{U_t}(v) = \begin{cases} v & \text{if } v \in S(\tilde{\theta}_t, \delta_t) \\ v* & \text{otherwise,} \end{cases}$$

where $v*$ denotes the closest point to v in the ball $S(\vartheta_t, \delta_t)$.

There are three main problems arising concerning the behaviour of the estimating procedures of type (15): the global convergence, that is the convergence of (15) for any starting point $\hat{\theta}_0$; the rate convergence; and the asymptotic distribution.

Note that in the case of an auxiliary consistent estimator the procedure (15) is automatically globally convergent. In general, given that usual regularity conditions are satisfied (e.g., conditions similar to (13) and the appropriate rate of the normalising sequence), the construction of the procedure guarantees the local convergence. In other words, the estimator will converge to θ, provided that the values of the procedure "stay" in a sufficiently small neighbourhood of θ. To ensure the global convergence, one need to impose the conditions of the global type on the corresponding functions, e.g. the conditions that guarantee the property of type (13) for any u, and also conditions on the growth of the corresponding functions at infinity (see [3] for details). Once the convergence is secured, the rate of convergence and the asymptotic distribution depend on the local behaviour of the corresponding functions (like differentiability of higher order) and the ergodicity of the model (see [4]-[5]). For instance, when studying asymptotic distribution, the main task is to show that $\hat{\theta}_t$ is locally asymptotically linear, that is

$$\hat{\theta}_t = \theta + \Gamma_t^{-1}(\theta) \sum_{s=1}^{t} \psi_s(\theta) + \rho_t^{\theta}, \tag{16}$$

and $\Gamma_t^{1/2}(\theta)\rho_t^{\theta} \to 0$ in probability. Asymptotic distribution of an asymptotically linear estimator can be studied using a suitable form of the central limit theorem.

3 Examples

Example 1. ***AR(1) - Linear procedures*** Suppose that X_t is an AR(1) process defined by (1) where ξ_t is a martingale-difference, that is, $E_\theta\{\xi_t \mid \mathcal{F}_{t-1}\} = 0$. Suppose also that $D_t = E_\theta\left\{\xi_t^2 \mid \mathcal{F}_{t-1}\right\} > 0$

and consider the recursive estimator

$$\hat{\theta}_t = \hat{\theta}_{t-1} + \frac{1}{\hat{\Gamma}_t D_t} X_{t-1} \left(X_t - \hat{\theta}_{t-1} X_{t-1} \right) \tag{17}$$

where

$$\hat{\Gamma}_t = \hat{\Gamma}_{t-1} + \frac{X_{t-1}^2}{D_t}. \tag{18}$$

Proposition *Suppose that*

$$\hat{\Gamma}_t = \hat{\Gamma}_0 + \sum_{s=1}^{t} \frac{X_{s-1}^2}{D_s} \ \rightarrow \ \infty.$$

Then the estimator $\hat{\theta}_t$ defined by (17) *is strongly consistent, that is, $\hat{\theta}_t \rightarrow \theta$ a.s. for any initial values $\hat{\theta}_0$ and $\hat{\Gamma}_0$. Furthermore, if*

$$\lim_{t \rightarrow \infty} \frac{\triangle \hat{\Gamma}_t}{\hat{\Gamma}_{t-1}} = 0, \tag{19}$$

a.s., then $\hat{\Gamma}_t^{\delta} |\hat{\theta}_t - \theta| \rightarrow 0$ a.s. for any $\delta \in]0, 1/2[$ and for any initial values of $\hat{\theta}_0$ and $\hat{\Gamma}_0$.

Proof The proof is given in [6].

In the case of the i.i.d. innovations ξ_t we have $\hat{\Gamma}_t = \hat{\Gamma}_0 + \text{const} \sum_{s=1}^{t} X_{s-1}^2 \rightarrow \infty$ for any θ, implying that in this case $\hat{\theta}_t$ is strongly consistent for any value of the parameter θ. Also, it is easy to see that (19) holds if e.g., the limit $\hat{\Gamma}_t / t$ exists (a.s.) and is finite. For example, in the case of the i.i.d. innovations ξ_t, this will happen if $|\theta| \leq 1$ implying that $t^{\delta} |\hat{\theta}_t - \theta| \rightarrow 0$ a.s. for any $\delta \in]0, 1/2[$.

Example 2. ***AR(1) - Likelihood procedures*** Let X_t be strongly stationary and let ξ_t be i.i.d. and independent from X_0. Suppose that g is the common probability density function of ξ_t. Consider the recursive estimator $\hat{\theta}_t$ defined by

$$\begin{aligned} \hat{\theta}_t &= \left[\hat{\theta}_{t-1} - K \hat{I}_t^{-1} X_{t-1} \frac{g'(X_t - \hat{\theta}_{t-1} X_{t-1})}{g(X_t - \hat{\theta}_{t-1} X_{t-1})} \right]_{\alpha_t}^{\beta_t}, \\ \hat{I}_t &= \hat{I}_{t-1} + X_{t-1}^2, \end{aligned} \tag{20}$$

where K is any positive constant and (α_t, β_t) is a random truncation sequence with $-\infty \leq \alpha_t \leq \beta_t \leq \infty$ (-a.s.) and $\theta \in [\alpha_t, \beta_t]$ for large $t's$. If g is bell-shaped and symmetric about zero, and the function g'/g is bounded and continuous at zero, then $\hat{\theta}_t \rightarrow \theta$ a.s. for any staring value $\hat{\theta}_0$ (this and more general results can be found in [6]).

Let us now consider the recursive estimator with the LS truncations.

Proposition *Let X_t be strongly stationary and ξ_t be i.i.d. and independent from X_0. Suppose that ξ_t have a finite fourth moment and a common probability density function g. Consider the recursive estimator defined by* (6), (2) *and* (3), *where $1/4 \leq \varepsilon < 1/2$ and*

$$0 < i_g = \frac{d}{dw} \int_{-\infty}^{\infty} \frac{g'}{g} (z - w) \, g(z) dz \Big|_{w=0} < \infty. \tag{21}$$

Suppose also that for some $\varepsilon_0 > 0$

$$\int_{-\infty}^{\infty} \frac{g'}{g} (z - w) \, g(z) dz = -i_g w + w^{1+\varepsilon_0} O(1), \tag{22}$$

as $w \rightarrow 0$,

$$\int_{-\infty}^{\infty} \left[\frac{g'}{g}(z) \right]^2 g(z) < \infty, \tag{23}$$

and

$$\int_{-\infty}^{\infty} \left[\frac{g'}{g}(z - w) - \frac{g'}{g}(z) \right]^2 g(z) dz \rightarrow 0 \tag{24}$$

as $w \to 0$.

Then $t^\delta|\hat{\theta}_t - \theta| \to 0$ *a.s. for any* $\delta \in]0, 1/2[$ *and any starring values* $\hat{\theta}_0$. *Furthermore,* $\hat{\theta}_t$ *is asymptotically efficient in the sense that*

$$\mathcal{L}(\hat{I}_t^{1/2}(\hat{\theta}_t - \theta)) \xrightarrow{d} \mathcal{N}\left(0,\ i_g^{-1}\right), \tag{25}$$

and also,

$$\mathcal{L}(t^{1/2}(\hat{\theta}_t - \theta)) \xrightarrow{d} \mathcal{N}\left(0,\ \frac{(1-\theta^2)}{\sigma^2 i_g}\right)$$

where $\sigma^2 = var(\xi_t)$.

Proof The proof is given in [6].

Note that under usual regularity assumptions, $i^g = \int (g'(z)/g(z))^2 g(z)\, dz$, implying that $i_t = i_g X_{t-1}^2$ is the one step conditional Fisher information and the total conditional Fisher information is

$$I_t = i_g \sum_{s=1}^{t} X_{t-1}^2 = i_g \hat{I}_t.$$

So, (25) reflects the fact that $(\hat{\theta}_t - \theta)$ is asymptotically normal with asymptotic variance I_t^{-1}, where I_t is the conditional Fisher information.

Example 3. ***An explicit example - AR(1) with Student innovations*** Suppose that X_t is a strictly stationary and ξ_t are independent Student random variables with degrees of freedom α. So, the probability density functions of ξ_t is

$$g(z) = C_\alpha \left(1 + \frac{z^2}{\alpha}\right)^{-\frac{\alpha+1}{2}},$$

where $C_\alpha = \mathbf{\Gamma}((\alpha+1)/2)/(\sqrt{\pi\alpha}\,\mathbf{\Gamma}(\alpha/2))$. Since

$$\frac{g'(z)}{g(z)} = -(\alpha+1)\frac{z}{\alpha + z^2},$$

it is easy to see that the Fisher information is

$$i^g = \int \left(\frac{g'(z)}{g(z)}\right)^2 g(z)\, dz = \frac{\alpha+1}{\alpha+3}.$$

Consider a likelihood recursive procedure with $-\infty \le \alpha_t \le \beta_t \le \infty$:

$$\hat{\theta}_t = \left[\hat{\theta}_{t-1} + \hat{I}_t^{-1} i_g^{-1}(\alpha+1) X_{t-1} \frac{X_t - \hat{\theta}_{t-1} X_{t-1}}{\alpha + (X_t - \hat{\theta}_{t-1} X_{t-1})^2}\right]_{\alpha_t}^{\beta_t}, \quad t \ge 1, \tag{26}$$

where

$$\hat{I}_t = \hat{I}_{t-1} + X_{t-1}^2$$

and $\hat{\theta}_0$ is any starting point. If $\alpha \ge 3$, $\hat{\theta}_t$ is strongly consistent provided that $\theta \in (\alpha_t, \beta_t)$ for large t's, in particular when $\alpha_t = -\infty$ and $\beta_t = \infty$.

Now consider (26) with the LS truncations, that is, when

$$\alpha_t = \hat{\theta}_t^{LS} - c\hat{I}_t^{-\varepsilon} \text{ and } \beta_t = \hat{\theta}_t^{LS} + c\hat{I}_t^{-\varepsilon}. \tag{27}$$

It is not difficult to see that if $1/4 \le \varepsilon < 1/2$ and $\alpha \ge 5$, all the conditions of the proposition in the previous example are satisfied (all the improper integrals involved are uniformly convergent and all the corresponding functions are infinitely many times differentiable). Thus, if $\alpha \ge 5$, the recursive estimator defined by (26) and (27) with $1/4 \le \varepsilon < 1/2$, is strongly consistent with $t^\delta|\hat{\theta}_t - \theta| \to 0$ (a.s.) for any $\delta \in]0, 1/2[$.

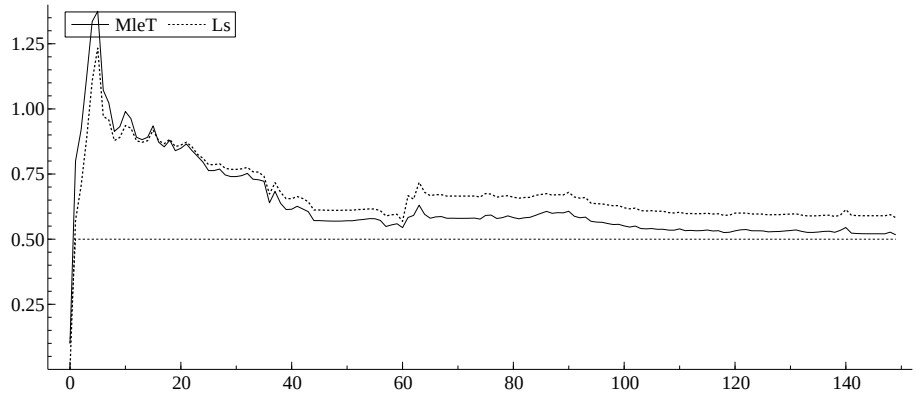


Figure 1: Realisations of $\hat{\theta}_t$ (MleT) and $\hat{\theta}_t^{LS}$ (Ls)

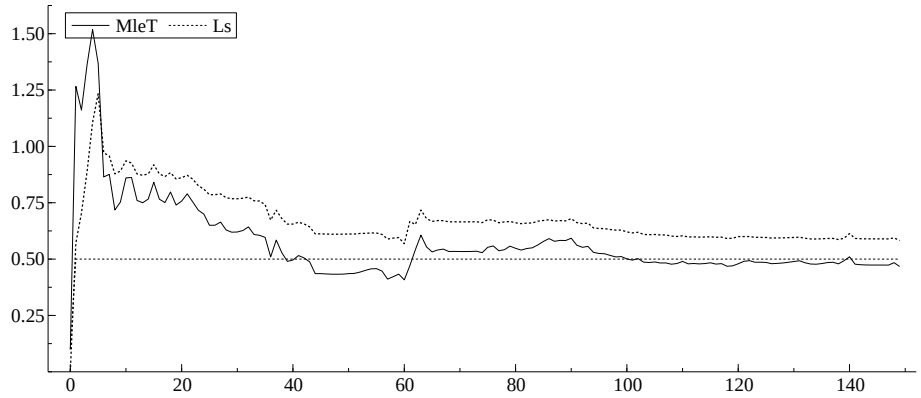


Figure 2: Realisations of $\hat{\theta}_t$ (MleT) and $\hat{\theta}_t^{LS}$ (Ls)

Furthermore, $\hat{\theta}_t$ is asymptotically efficient, i.e. $\mathcal{L}\left(\hat{I}_t^{1/2}(\hat{\theta}_t-\theta)\right)\xrightarrow{d}\mathcal{N}\left(0,\ i_g^{-1}\right)$, and also, $\mathcal{L}\left(t^{1/2}(\hat{\theta}_t-\theta)\right)\xrightarrow{d}\mathcal{N}\left(0,\ (1-\theta^2)\left(1-\frac{6}{\alpha}+\frac{6}{1+\alpha}\right)\right)$.

As far as the practical implementation of this procedure is concerned, it is important to note that the asymptotic behaviour of $\hat{\theta}_t$ will not change (including the rate of convergence), if we replace $\hat{I}_t$ in (26) (or, in general, in (20)) by $c_t\hat{I}_t$, where $c_t>0$ are constants with $c_t=1$ for large t's (see [6] for details). In practice, c_t can be treated as tuning constants to control behaviour of the normalising sequence for the first several steps, especially when the number of observations is small or even moderately large. As it was mentioned above, at each step, the recursive procedure (26) (or, in general (20)) on average moves towards the parameter. Nevertheless, if the values of the normalizing sequence are too small for the first several steps, then the procedure will oscillate excessively around the true value of the parameter. On the other hand, too large values of the normalizing sequence will result in delay of the estimator reaching the value of the parameter. A good balance can be achieved by using the tuning constants.

Figures 1 and 2 show realisations of the estimators $\hat{\theta}_t$ and $\hat{\theta}_t^{LS}$ for $t=0,\ldots,150$, when the observations are from AR(1) process with the iid Student innovations with $\alpha=5$, $\theta=0.5$, $\hat{\theta}_0=\hat{\theta}_0^{LS}=0.1$ and $\hat{I}_0=0$. $\hat{\theta}_t$ is derived from (26) with (27) truncations where $\varepsilon=1/4$ and $c=1$. As we can see from Figure 1, $\hat{\theta}_t$ is moving downwards slowly. This may be due to the high values of the nor-

malising sequence at the beginning of the procedure. Figure 2 shows the values of $\hat{\theta}_t$ for the same realisation but the normalising sequence $\hat{I}_t$ is replaced by $c_t\hat{I}_t$, where $c_t = 0.6$ for $t = 1, \ldots, 15$ and $c_t = 1$ otherwise. Now the path of the estimator has a "proper" shape, that is a reasonable oscillation at the beginning of the procedure before settling down at a particular level. On other occasions, it may be desirable to increase the values of the normalising sequence for the first several steps. This happens when the procedure oscillates too excessively before settling down at a particular level. This can be dealt with by introducing a positive constant $\hat{I}_0 \neq 0$ and/or setting the values of c_t greater than one for the first several values of the normalising sequence $c_t\hat{I}_t$.

References

[1] Fabian, V. (1978) On asymptotically efficient recursive estimation, *Ann. Statist.*, **6**, pp. 854-867.

[2] Khas'minskii, R.Z. & Nevelson, M.B. (1972) *Stochastic Approximation and Recursive Estimation*, Nauka, Moscow.

[3] Sharia, T. (2008) Recursive parameter estimation: Convergence. *Statistical Inference for Stochastic Processes*, **11**, 2, pp. 157 – 175.

[4] Sharia, T. (2007) Rate of convergence in recursive parameter estimation procedures. *Georgian Mathematical Journal*, Volume **14** (2007), 4, pp. 721–736.

[5] Sharia, T. (2008) Recursive parameter estimation: Asymptotic expansion (2008). *The Annals of The Institute of Statistical Mathematics* (DOI: 10.1007/s10463-008-0179-z).

[6] Sharia, T. (2008) New efficient estimation procedures in autoregressive time series models, *http://personal.rhul.ac.uk/UkAH/113/AR.pdf*

Variational Markov Chain Monte Carlo for Inference in Partially Observed Nonlinear Diffusions

Yuan Shen
Neural Computing Research Group
Aston University
Birmingham, United Kingdom
sheny2@aston.ac.uk

Cedric Archambeau
Department of Computer Science
University College London
London, United Kingdom
c.archambeau@cs.ucl.ac.uk

Dan Cornford
Neural Computing Research Group
Aston University
Birmingham, United Kingdom
d.cornford@aston.ac.uk

Manfred Opper
Artificial Intelligence Group
Technical University Berlin
Berlin, Germany
opperm@cs.tu-berlin.de

Abstract

In this paper, we develop set of novel Markov chain Monte Carlo algorithms for Bayesian inference in partially observed non-linear diffusion processes. The Markov chain Monte Carlo algorithms we develop herein use an approximating distribution to the true posterior as the proposal distribution for an independence sampler. The approximating distribution utilises the posterior approximation computed using the recently developed variational Gaussian Process approximation method. Flexible blocking strategies are then introduced to further improve the mixing, and thus the efficiency, of the Markov chain Monte Carlo algorithms. The algorithms are tested on two cases of a double-well potential system. It is shown that the blocked versions of the variational sampling algorithms outperform Hybrid Monte Carlo sampling in terms of computational efficiency, except for cases where multi-modal structure is present in the posterior distribution.

1 Introduction

Stochastic dynamical systems, also often referred to as diffusion processes or stochastic differential equations (SDEs), have been used for modelling of real-life systems in various areas ranging from physics to system biology to environmental science [1]. This work has been motivated by the problem of data assimilation [2] where such systems, representing the evolution of the atmosphere system are observed by an array of different instruments and the aim is inference of the current state of the system. Such continuous time systems are often only partially observed, which makes likelihood based statistical inference difficult. From a methodological point of view, the inference problem for stochastic dynamical systems has been pursued in three main directions.

The first direction is based on solving the Kushner-Stratonovich-Pardoux (KSP) equations [3] which are the most general optimal solutions to the inference problem. However, solution of the KSP equations is numerically intractable for high-dimensional non-linear systems, so various approximation strategies have been developed. In the particle filtering method [4], the solution of the KSP filtering equations, namely the posterior density, is approximated by a discrete distribution with random support. For linear, Gaussian systems, the filtering part of KSP equations reduces to the well-known Kalman-Bucy filter [5]. To treat non-linear systems a number of approximation strategies have extended the Kalman filter, for example, the ensemble Kalman filter [6], and unscented Kalman filter [7].

The second direction involves a variational approximation to the posterior process. In [8], a linear diffusion approximation is proposed and its linear drift is optimised globally. This is explained in more detail in Section 3. In [9], a mean field approximation is applied to the KSP equations and the mean field representation of possible trajectories is optimised globally.

The third direction employs Markov Chain Monte Carlo (MCMC) methods [10] to sample the posterior process, which is the focus of this paper. At each step of a MCMC simulation, a new state is proposed and will be accepted or rejected in a probabilistic way. For applications to stochastic dynamical systems, it is also often referred to as path sampling. A path sampling approach to discrete-time state-space models has been addressed in [11] and references therein. In those works, a Gibbs-sampler with single-site update was used. To achieve better mixing, two closely related MCMC algorithms for path sampling, namely the Metropolis-adjusted Langevin and the Hybrid Monte Carlo (HMC) algorithm, were recently proposed in [12] and [13], respectively. The both methods update the entire sample path at each sampling iteration while keeping the acceptance of new paths high. This is achieved by combining the basic MCMC algorithm with a fictitious dynamics so that the MCMC sampler proposes moves towards the regions of higher probability in the state space. Another strategy to achieve better mixing in path sampling is to update one of the subpaths between two neighbouring observations at each Metropolis-Hastings step leading to "blocking strategies". In [14], the so-called "modified diffusion bridge" approach is used to propose candidates for such sub-paths. The similar bridging method, suggested in [15], is based on a crude discretization scheme for the generating SDE. To the same end, a so-called "retrospective sampling" method is used in [16] which can simulate a wide class of diffusion bridge processes exactly, under certain conditions on the stochastic noise process.

In this paper, we present a novel MCMC algorithm for path sampling of non-linear diffusion processes. The new algorithm employs the variational approximation method in [8] to produce a more computationally efficient sampling method. Our MCMC algorithm also extends the blocking strategies in [14, 15], allowing blocks of arbitrary size. The idea of using the variational posterior distribution as the proposal distribution for MCMC samplers in not new [17] and is referred to as Variational MCMC. Our algorithm makes use of information from the data, which guides the Markov chain to make proposals from locations in the solution space which have considerable support under the posterior distribution. Precisely speaking, information from the data is encoded in the variational Gaussian process which approximates the true posterior process. Thus we might also consider our algorithm as being within the spirit of data-driven MCMC [18]. Further, the variational MCMC sampler can also be used in a setting of "mixture of transition kernels" [17], and the mixture kernel including both HMC and the variational sampler can be defined adaptively.

The paper is organised as follows; Section 2 first presents Bayesian treatment of non-linear smoothing which is followed in Section 3 by a summary of the variational Gaussian process smoother [8] that is used to provide the proposal density. The novel algorithms are described in Section 4 and the performance of these algorithms is demonstrated in Section 5 by numerical experiments with two variants of a double-well potential system. The paper concludes with a discussion.

2 Bayesian inference for non-linear diffusions

Mathematically, a stochastic dynamical system is often represented by a SDE [19]:

$$d\mathbf{x}(t) = \mathbf{f}(\mathbf{x}, t)dt + \mathbf{D}^{1/2}(\mathbf{x}, t)d\mathbf{W}(t), \tag{1}$$

where $\mathbf{x}(t) \in \mathcal{R}^d$ is the state vector, $\mathbf{D} \in \mathcal{R}^{d\times d}$ is the so-called diffusion matrix, and $\mathbf{f}$ represents a deterministic dynamical process, generally called the drift. The driving noise process is represented by a Wiener process $\mathbf{W}(t)$. (1) is also referred to as a diffusion process. The state is observed via some measurement function $\mathbf{h}(\cdot)$ at discrete times, say $\{t_k\}_{k=1,...,M}$. The observations are assumed contaminated by i.i.d Gaussian noise:

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}(t_k)) + \mathbf{R}^{\frac{1}{2}} \cdot \eta \tag{2}$$

where $\mathbf{y}_k \in \mathcal{R}^{d'}$ is the k-th observation, $\mathbf{R} \in \mathcal{R}^{d'\times d'}$ is the covariance matrix of measurement errors, and η represents multivariate white noise.

A Bayesian approach to smoothing is typically adopted in which the posterior distribution

$$p(\mathbf{x}([0, T])|\{\mathbf{y}_1, ..., \mathbf{y}_M, t_M < T\}),$$

is formulated and estimated, using for example the methods described in Section 1. In this work, we discretise the continuous-time SDE, using an explicit Euler-Maruyama scheme [19], and thus treat an approximate non-linear discrete time model which describes a Markov chain. The discretized version of (1) is given by

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{f}(\mathbf{x}_k, t_k)\delta t + \mathbf{D}^{1/2}(\mathbf{x}_k, t_k)\sqrt{\delta t} \cdot \xi_k, \tag{3}$$

with $t_k = k \cdot \delta t$, k = 0, 1, ..., N, and a smoothing window from t = 0 to $T = N \cdot \delta t$. Note that ξ_k are white noise random variables. An initial state, $\mathbf{x}_0$, needs to be set. There are M observations within the smoothing window chosen at a subset of discretisation times $(t_{k_j}, \mathbf{y}_j)_{j=1,...,M}$ with $\{t_{k_1}, ..., t_{k_M}\} \subseteq \{t_0, ..., t_N\}$. In the following, we formulate the posterior distribution step by step. The prior of a diffusion process, exploiting the Markov property, can be written as

$$p(\mathbf{x}_0, ..., \mathbf{x}_N) = p(\mathbf{x}_0) \cdot p(\mathbf{x}_1|\mathbf{x}_0) \cdot .... \cdot p(\mathbf{x}_N|\mathbf{x}_{N-1}),$$

where $p(\mathbf{x}_0)$ is the prior on the initial state and $p(\mathbf{x}_{k+1}|\mathbf{x}_k)$ with k = 0, ..., $N-1$ are the transition densities of the diffusion process. In the limit of small enough δt, those transition densities can be well approximated by a Gaussian density [20] and thus $p(\mathbf{x}_{k+1}|\mathbf{x}_k) = \mathcal{N}(\mathbf{x}_{k+1}|\mathbf{x}_k + f(\mathbf{x}_k)\delta t, \mathbf{D}\delta t)$. Therefore, the prior over the path, defined by the SDE is given by

$$p(\mathbf{x}_0, ..., \mathbf{x}_N) \propto p(\mathbf{x}_0) \cdot \exp(-\mathcal{H}_{\text{dynamics}}),$$

where

$$\mathcal{H}_{\text{dynamics}} = \sum_{k=0}^{N-1} \frac{\delta t}{2} \left[\frac{\mathbf{x}_{k+1} - \mathbf{x}_k}{\delta t} - \mathbf{f}(\mathbf{x}_k, t_k)\right]^\top \mathbf{D}^{-1} \left[\frac{\mathbf{x}_{k+1} - \mathbf{x}_k}{\delta t} - \mathbf{f}(\mathbf{x}_k, t_k)\right].$$

Assuming the measurement noise is i.i.d. Gaussian, the likelihood is simply given by

$$p(\mathbf{y}_1, ..., \mathbf{y}_M|\mathbf{x}(t_0), ..., \mathbf{x}(t_N)) \propto \exp(-\mathcal{H}_{\text{obs}}),$$

where

$$\mathcal{H}_{obs} = \frac{1}{2} \sum_{j=1}^{M} \left[\mathbf{h}(\mathbf{x}(t_{k_j})) - \mathbf{y}_j\right]^\top \mathbf{R}^{-1} \left[\mathbf{h}(\mathbf{x}(t_{k_j})) - \mathbf{y}_j\right]. \tag{4}$$

In summary, we have the posterior distribution given by

$$p(\mathbf{x}(t)|\{\mathbf{y}_1, ..., \mathbf{y}_M\}) \propto p(\mathbf{x_0}) \cdot \exp(-(\mathcal{H}_{\text{dynamics}} + \mathcal{H}_{\text{obs}})).$$

3 Variational Gaussian process approximation smoother

The starting point of the variational Gaussian process approximation method [8] is to approximate (1) by a linear SDE:

$$d\mathbf{x}(t) = \mathbf{f}_L(\mathbf{x}, t)dt + \mathbf{D}^{1/2}(\mathbf{x}, t)d\mathbf{W}(t), \tag{5}$$

where the time varying linear drift approximation is given by

$$f_L(\mathbf{x}, t) = -\mathbf{A}(t)\mathbf{x}(t) + \mathbf{b}(t). \tag{6}$$

The matrix $\mathbf{A}(t) \in \mathcal{R}^{d \times d}$ and the vector $\mathbf{b}(t) \in \mathcal{R}^d$ are two variational parameters to be optimised.

The approximation made in (6) implies that the true posterior process, i.e. $p(\mathbf{x}(t)|\mathbf{y}_1, ..., \mathbf{y}_M)$, is approximated by a Gaussian Markov process, $q(t)$. If we discretise the linear SDE in the same way as the true SDE, the approximate posterior can be written down as

$$q(\mathbf{x}_0, ..., \mathbf{x}_N) = q(\mathbf{x}_0) \cdot \prod_{k=0}^{N-1} \mathcal{N}(\mathbf{x}_{k+1}|\mathbf{x}_k + f_L(\mathbf{x}_k)\delta t, \mathbf{D}\delta_t).$$

Note that $q(\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_0|m(0), S(0))$. The optimal $\mathbf{A}(t)$ and $\mathbf{b}(t)$, together with the optimal marginal means and covariances $\mathbf{m}(t)$ and $\mathbf{S}(t)$, are obtained by minimising the KL divergence of $q(\cdot)$ and $p(\cdot)$ [8]. With the estimated $\mathbf{A}(t)$ and $\mathbf{b}(t)$, we are able to obtain the two-time covariance function $K(t_1, t_2)$ using

$$\frac{d\mathbf{K}(t_1, t_2)}{dt_1} = -\mathbf{A}(t_1)\mathbf{K}(t_1, t_2) \tag{7}$$

for $t_1 > t_2$ with $\mathbf{K}(t_1, t_2) = \mathbf{K}(t_2, t_1)$ for $t_2 > t_1$. We note that the variational approximation can also be derived in continuous time using Girsanov's change of measure theorem, however it is then necessary to discretise the system for computational implementation [21].

4 Variational MCMC methods

In a Metropolis-Hastings algorithm [22] for sampling a posterior density $\pi(x)$, defined on a general state space $\mathcal{X}$, one proposes a new state $x' \in \mathcal{X}$ according to some density $q(x, x')$. The proposed state will be accepted with probability $\alpha(x, x')$, given by

$$\alpha = \min\left\{1, \frac{\pi(x')}{\pi(x)} \cdot \frac{q(x', x)}{q(x, x')}\right\}.$$

When the Metropolis-Hastings (MH) algorithm is applied to a particular Bayesian inference problem, intelligent proposal mechanisms are often required to make the algorithm efficient.

In our variational MCMC algorithms, we make proposals using $\mathbf{A}$, $\mathbf{b}$, $\mathbf{m}$, $\mathbf{S}$, and $K(\cdot,\cdot)$ estimated by applying the variational Gaussian process method described in Section 3, to the data set we consider. To implement the algorithm, we employ an independence sampler in which any proposal is independent of the current state. In the following, we describe two different implementations:

"Variational multivariate sampler" makes proposals x' by sampling from the multivariate Gaussian distribution with the mean $\mathbf{m}$ and covariance function $K(\cdot,\cdot)$. This means that

$$\mathbf{x}' = \mathbf{m} + \mathbf{L}^\top \mathbf{w},$$

where $\mathbf{L}$ is the Cholesky decomposition of $\mathbf{K}$ and $\mathbf{w} = (w_0, ..., w_N)^\top$ is a vector of white noise;

"Variational simulation sampler" makes proposals by integrating the approximate linear SDE with the drift term given by $\mathbf{f}_\mathbf{L}(\mathbf{x}, t) = -\mathbf{A}(t) + \mathbf{b}(t)$. The initial value $\mathbf{x}_0$ is sampled from $\mathcal{N}(\mathbf{x}_0|\mathbf{m}(0), \mathbf{S}(0))$. This means that

$$\mathbf{x}_0 = \mathbf{m}(0) + \sqrt{\mathbf{S}(0)} \cdot w_0$$

and

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{f}_\mathbf{L}(\mathbf{x}, t)\delta t + \mathbf{D}^{1/2}\sqrt{\delta t} \cdot w_k$$

with k = 1, ..., N.

In both cases, $N + 1$ random numbers are sampled from the standard Gaussian distribution independently. Therefore, the log proposal probability, up to a normalising constant, can be calculated by $-\mathbf{w}^\top \mathbf{w}/2$.

The efficiency of the independence sampler depends on how well the proposal density approximates the target measure. In this work, our proposal density is a Gaussian process approximation to the target measure. Moreover, the approximate density is optimised with respect to the first and second moment of the target measure. Therefore, the efficiency of the above algorithms is determined by how far the target measure deviates from a Gaussian one. In cases with a highly non-linear drift term in the diffusion *and* relatively few observations, the above proposal mechanisms need to be further refined.

The idea of blocking helps to improve the performance of our independence samplers. Simply speaking, we propose only a part of the sample path at each MH step while the remaining parts are fixed, and conditioned on. To implement blocking, the whole sample path is sub-divided into a block of size $l + 1$,

$$B_1 = \{x_0, x_1, ..., x_l\}$$

and $M - 1$ blocks of size l, say

$$B_k = \{x_{(k-1)*l+1}, ..., x_{k*l}\}$$

with k = 2, ..., M. This means that $N = M \cdot l$. At each MH step, one block is chosen at random and a proposal is made for the sub-path within this block by conditional sampling. The conditional sampling versions of the algorithms are described in the following:

"Block variational multivariate sampler" carries out the conditional sampling of block k, first using a permutation matrix $\mathcal{P}$ so that

$$\mathcal{P}(B_1, ..., B_M)^\top = (B_k, B_1, ..., B_{k-1}, B_{k+1}, B_M)^\top,$$

$\hat{\mathbf{m}} = \mathcal{P}\mathbf{m}$, and $\hat{K} = A\mathbf{K}A^\top$. After the permutation, the first block can be sampled, i.e. $y = B_k^\top$, conditioning on the remaining blocks

$$x = (B_1, ..., B_{k-1}, B_{k+1}, B_m)^\top$$

from the Gaussian distribution $\mathcal{N}(y|\hat{\mathbf{m}}_\mathbf{k}, \hat{\mathbf{K}}_k)$ using the following relations:

$$\hat{\mathbf{m}} = \begin{pmatrix}\mu_y \\ \mu_x\end{pmatrix} \quad \text{and} \quad \hat{\mathbf{K}} = \begin{pmatrix}\Sigma_{yy} & \Sigma_{yx} \\ \Sigma_{xy} & \Sigma_{xx}\end{pmatrix}$$

$$\hat{\mathbf{m}}_\mathbf{k} = \mu_y + \Sigma_{yx}\Sigma_{xx}^{-1}(x - \mu_x)$$

$$\hat{\mathbf{K}}_k = \Sigma_{yy} - \Sigma_{yx}\Sigma_{xx}^{-1}\Sigma_{xy} \ .$$

"Block variational simulation sampler" carries out the conditional sampling of block k by simulate a bridging process with two fixed ends at $t_{(k-1)*l}$ and t_{k*l+1}. In the following, we derive the effective drift and diffusion term of a SDE for such a process so that it can be easily simulated.

For clarity, we now consider a one-dimensional SDE and further reduce the problem of simulating a bridging process to the following question: how to sample x_t at time t with $t = t' + \delta t$ and $t < T$ conditioning $x_{t'}$ at time t' and x_T at time T? Note that $\Delta t = T - t' >> \delta t$.

To sample x_t, first compute the conditional probability

$$p(x_t|x_{t'}, x_T) \propto p(x_t|x_{t'}) \cdot p(x_T|x_t) \ .$$

As the time increment δt is the one used to discretise both the original and approximate SDE by a Euler-Maruyama scheme

$$p(x_t|x_{t'}) \propto \exp\left\{\underbrace{-\frac{1}{2D\delta t}(x_t - x_{t'} - f_L(x_{t'})\delta t)^2}_{I_1}\right\} \ . \tag{8}$$

On the other hand, $p(x_T|x_t)$ can be expressed by the marginal density of x from the backward version of the approximate linear SDE in (5) which is now initialised with x_T at time T. This means that

$$p(x_T|x_t) \propto \exp\left\{\underbrace{-\frac{1}{2\mathbf{d}_t}(x_t - \mathbf{c}_t)^2}_{I_2}\right\} \tag{9}$$

where $\mathbf{c}_t$ and $\mathbf{d}_t$ are obtained by integrating the following two ordinary differential equations backwards in time:

$$\frac{d}{dt}\mathbf{c}_t = -A\mathbf{c}_t + b$$

and

$$\frac{d}{dt}\mathbf{d}_t = -2A\mathbf{d}_t - D \ ,$$

which are actually the moment equations corresponding to the backward version of the linear approximate SDE. The initial values are $\mathbf{c}_T = x_T$ and $\mathbf{d}_T = 0$.

By re-formulating I_1 and I_2 in (8) and (9), respectively, the effective drift and diffusion terms for the bridging process are obtained as follows:

$$f_L^{eff} = -1 \cdot \underbrace{\frac{\mathbf{d}A + D}{\mathbf{d} + D\delta t}}_{A_t^{eff}} \cdot x_{t'} + \underbrace{\frac{\mathbf{c}D + b\mathbf{d}}{\mathbf{d} + D\delta t}}_{b_t^{eff}} \tag{10}$$

and

$$D_t^{eff} = D \cdot \frac{\mathbf{d}}{\mathbf{d} + D\delta t} \ . \tag{11}$$

In the continuous-time limit, the effective drift is given by

$$f_L^{eff} = f_L + \sigma^2 \partial_x \ln p(X_T|x_t)$$

while the effective diffusion term is D.

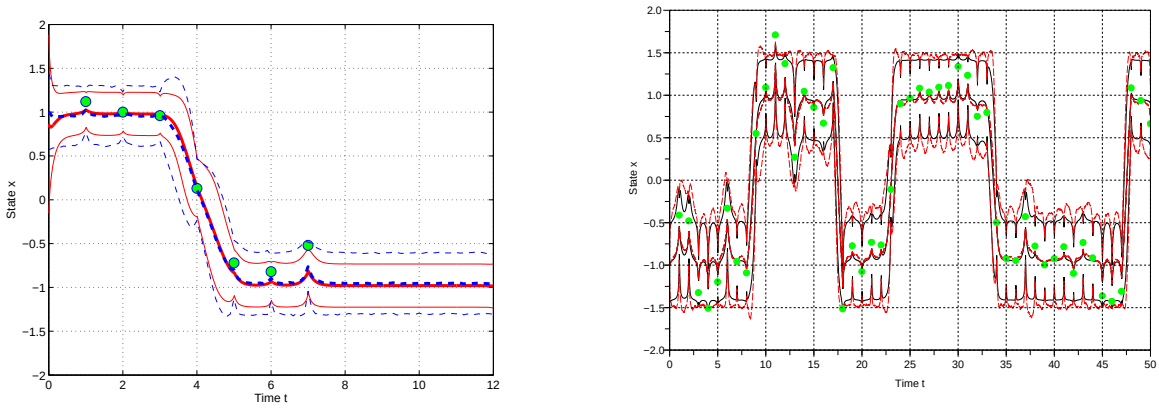


Figure 1: Comparison of the mean path and marginal variance estimates between the HMC (dashed) and variational (solid) method for two double-well potential systems, with diffusion variance κ^2 = 0.25 (left) and κ^2 = 1.0 (right). For each case, filled circles represent the observations, with measurement noise variance equal to 0.04. The mean paths are displayed by thick lines, while each pair of thin lines indicates an envelope of mean path with 2 $\times$ standard deviation.

5 Numerical Experiments

In this section, we compare the variational MCMC algorithms described in Section 4 with the state-of-the-art Hybrid Monte-Carlo (HMC) method based on the implementation developed in [12]. In HMC, the proposals for path $\mathbf{X} = (\mathbf{x}_0, ..., \mathbf{x}_N)$ are made by simulating a fictitious deterministic system as follows

$$\frac{d\mathbf{X}}{d\tau} = \mathbf{P} \quad \text{and} \quad \frac{d\mathbf{P}}{d\tau} = -\nabla_{\mathbf{X}} \hat{\mathcal{H}}(\mathbf{X}, \mathbf{P})$$

where τ is fictitious time, $\mathbf{P} = (\mathbf{p}_0, ..., \mathbf{p}_N)$ represents momentum, and $\hat{\mathcal{H}}$ is a fictitious Hamiltonian which is the sum of $-\log$ posterior probability of $\mathbf{X}$, i.e. $p(\mathbf{X}|\{\mathbf{y}_1, ..., \mathbf{y}_M\})$ defined in Sec. 2, and kinetic energy $\mathcal{H}^{kin} = \frac{1}{2}\sum_{k=1}^{N} \mathbf{p}_k^2$. The above system is initialised by setting $\mathbf{X}(\tau = 0) = \mathbf{X}$ (current path) and drawing $\mathbf{P}(\tau = 0)$ from a standard multivariate Normal distribution. After that, it is integrated forward in time with time increment $\delta\tau$ by using leapfrog [12]. A reasonably good mixing can be achieved by tuning the parameter $\delta\tau$ and the number of integration steps. Compared to the algorithm implemented in [12], the preconditioning matrix is not used here. A similar algorithm called Metropolis-adjusted Langevin (MALA) method is applied to path sampling in [13]. In contrast to HMC, the proposals for path $\mathbf{X}$ in MALA are made by integrating a SDE whose drift term is specified through the gradient of $-\log p(\mathbf{X}|\{\mathbf{y}_1, ..., \mathbf{y}_M\})$ (so-called Langevin equation).

We compare the different MCMC algorithms by their mixing properties and their burn-in period, since these are critical measures for the computational efficiency of a given MCMC algorithm. Mixing is measured by the auto-correlation function of the fictitious, algorithm induced time series of some summary statistic of a sample path. In this work, we look at both instantaneous values of the state $x(t)$ at different discrete times t and the summary statistic L, defined by $L = \int_{\mathcal{W}} x(t)dt$ where $\mathcal{W}$ denotes the smoothing window to assess mixing.

The algorithms are tested on two versions of a one-dimensional double-well potential systems. The double-well system is defined by

$$\dot{x}(t) = 4x(1 - x^2) + \kappa\xi(t) \ ,$$

where κ^2 is the diffusion variance and $\xi(t)$ is white-noise. This system has two stable states, namely $x = +1$ and $x = -1$ [23], and is often taken as an analogue for a system that has two stable states with atypical transitions between the two state, for example the climate of glacial and inter-glacial epochs. Depending on the value of κ, average times τ needed for the system to escape from one well can vary over a wide range of magnitudes. In this work, double-well systems with two different κ-values are considered. For κ = 0.5, τ amounts to about 3,000 time units, which makes a transition

unlikely within the smoothing window $\mathcal{W} = [0, 12]$ we choose. In contrast, the escape time for κ = 1.0 is about 8 time units. To allow for multiple transitions, we choose for this case a larger smoothing window, that is, $\mathcal{W} = [0, 50]$.

Further, we assume that the state x can be observed directly corrupted by additive Gaussian noise, and the observation error variance is chosen to be 0.04. For our numerical experiments, we make one observation per time unit for both cases. Accordingly, we generate two data sets, say data set A of 7 data points with κ = 0.5 and data set B of 50 data points with κ = 1.0.

The observations and a second order summary of the posterior paths for the HMC and variational Gaussian process approximation are shown in Figure 1. When applying the variational method to double-well systems, we discretise the SDE (5) with time increment δt = 0.01. For the MCMC

$t = 0$

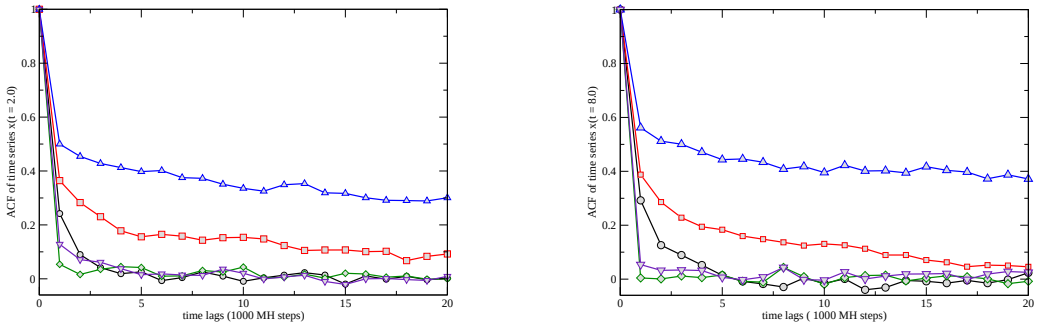


Figure 2: Comparison of the auto-correlation function of the time series of x at t = 2.0 (left) and that of x at t = 8.0 (right) between HMC (circle), the Variational Multivariate Sampler (box), the Block Variational Multivariate Sampler (diamond), the Variational Simulation Sampler (up-triangle) and the Block Variational Simulation Sampler (down-triangle) for data set A.

The initial experimental results focus on contrasting the four different variational MCMC algorithms with each other and the HMC results. In this case, data set A is considered. As can be seen from the description of the algorithm, the block variational multivariate sampler involves the inversion of large matrices. This is very time-consuming. Therefore, at this stage, we discretise the double-well potential model with δt = 0.1 for the MCMC algorithms while the variational Gaussian process method is applied with δt = 0.01. For the variational simulation sampler, we therefore need to coarsen the time resolution of $\mathbf{A}(t)$ and $\mathbf{b}(t)$ by sub-sampling the original ones. For those variational multivariate sampler, however, we first calculate $\mathbf{K}(t_1, t_2)$ with the fine-scale $\mathbf{A}(t)$ and $\mathbf{b}(t)$ and then sub-sample $\mathbf{K}$ accordingly. This ensures that the coarsening won't impair the algorithm

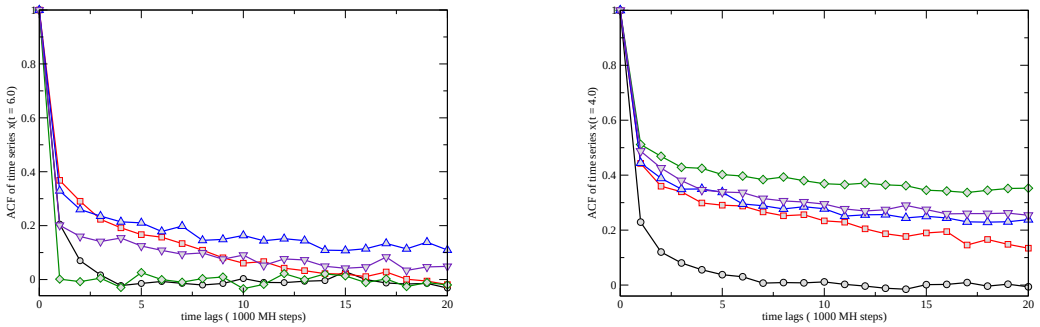


Figure 3: As Figure 2 but for time series of x at t = 6.0 (left) and that of x at t = 4.0 (right)

To compare different MCMC algorithms according to their mixing property, we calculate the auto-correlation functions of x-traces at fixed t = 2.0, t = 4.0, t = 6.0, and t = 8.0. For each algorithm, those traces are sub-sampled from the Markov chain of length equal to 5,000,000, with sampling interval equal to 1,000, after the burn-in period is discarded. The auto-correlation

functions are plotted against the number of sampling intervals in Figure 2 and Figure 3. From these figures, we can see that the blocking strategy greatly improves the performance of a naive independence sampler, except for the times t in the transitional phase (from t = 3.0 to t = 4.0). Outside this transition phase, it is also seen that both block variational samplers outperform the HMC approach. This means that the variational Gaussian process method can represent the information from the data in a convenient and efficient manner to guide a MCMC sampler by finding promising proposal distributions. The difficulties that occur in the transitional phase can be understood by the fact that the approximation made by variational Gaussian process to the true posterior process is necessarily relatively poor when the process shows multi-modal probability structure. Such multi-modal structure can be seen in a double-well system with a κ = 0.5 when the observations support probability in both modes of the system at transition times.

It is also clear from the algorithms that the simulation sampler is computationally much more efficient than the multivariate sampler. However, we notice that a simulation sampler generally performs worse, as measured by mixing, than its corresponding multivariate sampler. This can be explained by the coarsening of $\mathbf{A}(t)$ and $\mathbf{b}(t)$ in the simulation sampler. Therefore, the results of a multivariate sampler indicate the performance which the corresponding simulation sampler can achieve when fine time increments are used. By the same argument, it is also observed that the effect of coarsening is much less significant when the blocking strategy is adopted. This indicates that blocking can also help reduce the computational complexity by utilising a relatively coarse time resolution. Therefore, we now focus on the comparison between HMC and the block variational simulation sampler.

In the second part of the numerical experiments, we apply both the HMC and block variational simulation sampler to data sets A and B, with δt = 0.01. The histograms of state x at different times t are compared between these two MCMC algorithms, Fig 6 for the data set A with κ = 0.5 and Fig 7 for the data set B with κ = 1.0. To within sampling variability, both MCMC methods produce the same marginal sample distribution of state x.

To take CPU time into account in the comparison, we generate two long Markov chains with approximately same CPU time, one by HMC and another one by the variational sampler. Note that these two chains are of different length as the computational cost of a single MH step is different for those two sampler. The CPU time used in the experiment is about 2 hours for the data set with κ = 0.5 and about 28 hours for the data set with κ = 1.0. On the other hand, we sub-sample from these two chains with appropriate subsampling intervals so that 5, 000 sample paths are obtained from each chain. In the next step, we compare the burn-in period and mixing property of these two sequences of sample paths.

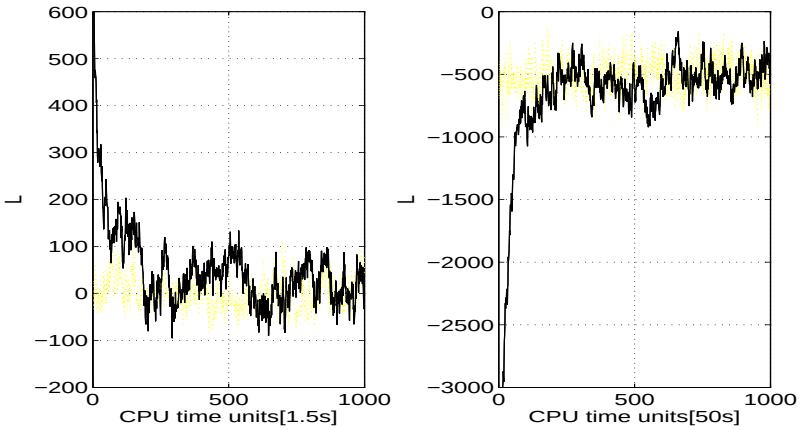


Figure 4: Comparison of burn-in period between HMC and the block variational simulation sampler for two data sets in Figure 1, with diffusion variance κ^2 = 0.25 (left) and κ^2 = 1.0 (right). In each panel, two traces of the summary statistic L, defined by $L = \int_{\mathcal{W}} x(t)dt$, are plotted for HMC (solid) and the block variational simulation sampler (dotted).

We compare the results in terms of both the burn-in indicated by the trace of summary statistic L and the auto-correlation of L. From Fig. 4 we can see that the burn-in of HMC is significantly longer than that of the variational MCMC method. This is partly because the variational MCMC make proposals by sampling from an approximate equilibrium distribution while HMC's proposals are always based on the current state. In addition, the blocking strategy helps to increase the acceptance of independent proposals made by the variational sampler. In Fig. 5, the decay of two auto-correlation functions of L-traces is compared. While the results for the data set with κ = 1.0 clearly show that our variational MCMC sampler has better mixing than HMC, the results for κ = 0.5 are not clear. It is evident that our method has fast decay of the auto-correlation function at short time scales but both methods are actually comparable at large time scales. This is maybe related to the observation that the variational MCMC sampler could occasionally get stuck for a long period of algorithm time, if a very probable state is accepted.

To summarise, the numerical experiments have demonstrated that the proposed blocking strategy has greatly improved the performance of two independence samplers. It is shown that the sample paths obtained from the block variational simulation sampler and those from HMC show very similar marginal distributions at different times t. We also find that the burn-in period needed for the variational sampler can be neglected, when compared to HMC. Finally, it is observed that the variational sampler provides better mixing than HMC in one example while both methods have comparable mixing in another example, which is very challenging for our variational Gaussian process approximation.

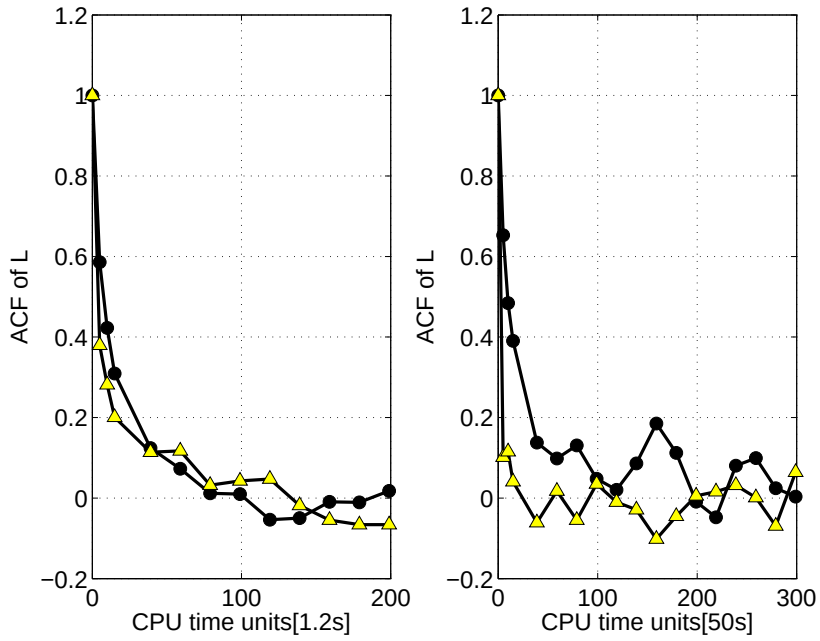


Figure 5: Comparison of mixing between HMC and the block variational simulation sampler for two data sets in Figure 1, with diffusion variance κ^2 = 0.25 (left) and κ^2 = 1.0 (right). In each panel, the ACF of two traces of the summary statistic L, defined by $L = \int_{\mathcal{W}} x(t)dt$, are plotted for HMC (circles) and the block variational simulation sampler (triangles).

6 Discussion

In this paper, we have presented two novel MCMC algorithms which both combine a particular MH algorithm, namely an independence sampler, and the variational Gaussian process approximation for Bayesian inference in non-linear diffusions. This demonstrates that variational approximations can be combined with MCMC methods to good effect. We stress that the variational approximation

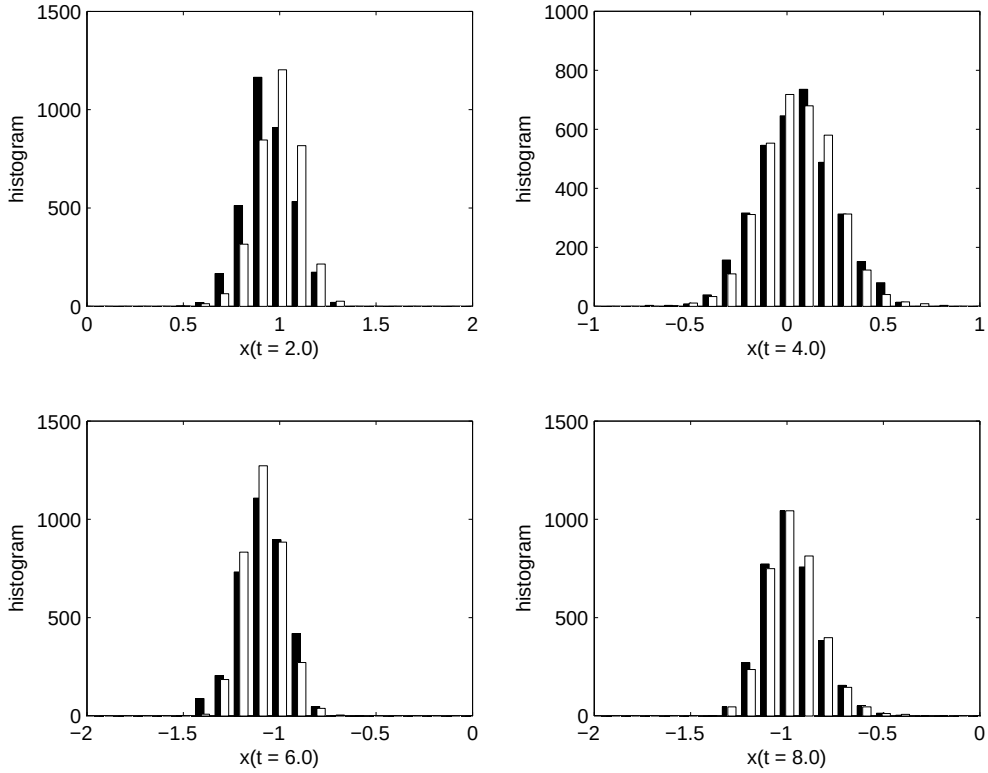


Figure 6: Comparison of histograms of state x at time t between HMC (filled) and the block variational simulation sampler (empty) for a double-well potential system with diffusion variance $\kappa^2 = 0.25$. Upper-left: $t = 2.0$; Upper-right: $t = 4.0$; Lower-left: $t = 6.0$; and Lower-right: $t = 8.0$.

introduced in this paper is not the traditional fully factorising approximation, rather the approximation is over the joint distribution of the state path, which make the variational approximation an attractive proposal distribution. Further, we introduce a flexible blocking strategy to improve the performance of the variational samplers.

One of our novel algorithms is based on conditional sampling of a multivariate Gaussian distribution whose mean and covariance function is obtained from the variational Gaussian process approximation. It has been shown that this algorithm outperforms Hybrid Monte Carlo, with one exception which we discuss below. However, the conditional sampling method involves inverting large matrices, which makes this algorithm time-consuming in practice, although this could be readily improved with some minor changes to the implementation and the exploitation of sparsity in the inverse covariance matrices.

The second algorithm makes proposals by simulating a bridge process of the approximate linear diffusion obtained from the variational Gaussian process approximation and thus this algorithm is very computationally efficient. Roughly speaking, it is 15 times faster than the optimised HMC we have implemented, which we believe is one of the most computationally efficient methods currently available for the problems we have tackled. Compared to the work presented in [15] and [14], we have adopted a very similar strategy for proposing a diffusion bridge process. Both [15] and [14] made a relatively crude Gaussian approximation to the marginal density of the process. In contrast, we exploit the more sophisticated approximation to the apparent, posterior, linear diffusion derived from the variational Gaussian process. In doing so, we integrate the corresponding moment equations backwards in time which may cause lower acceptance rates for the MCMC algorithm because the true *posterior process* may not time-reversible, however in experiments we do not see evidence for this having a practically noticeable effect, and we would expect that our variational samplers are significantly more efficient than those based on crude approximations.

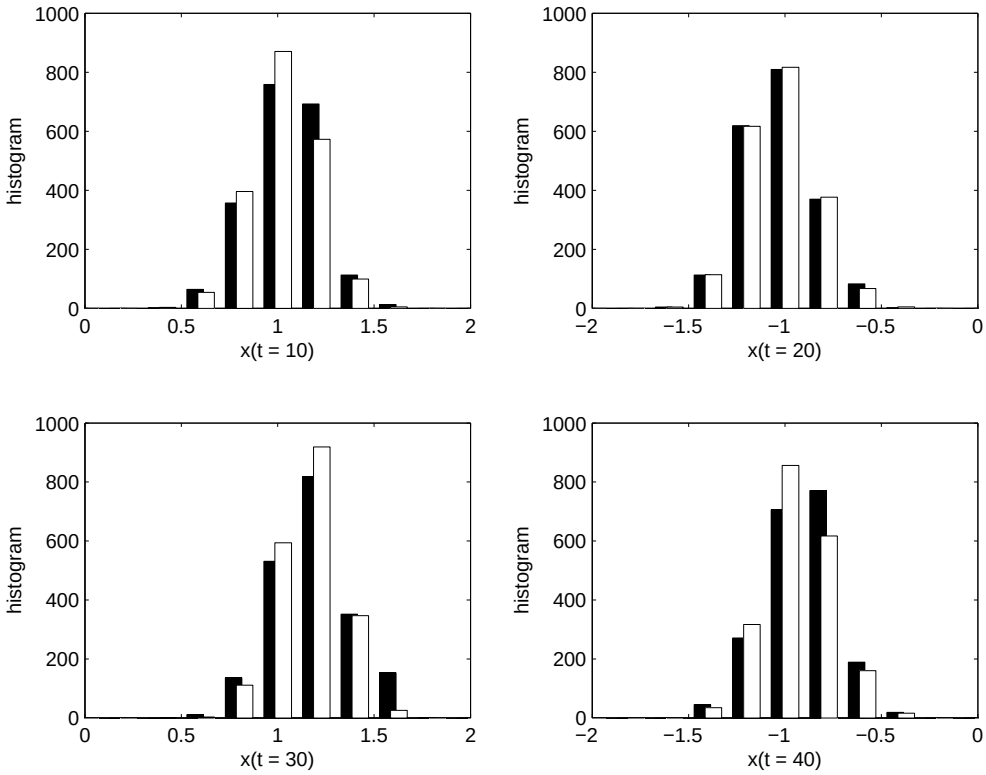


Figure 7: Comparison of histograms of state x at time t between HMC (filled) and the block variational simulation sampler (empty) for a double-well potential system with diffusion variance κ^2 = 1.0. Upper-left: t = 10; Upper-right: t = 20; Lower-left: t = 30; and Lower-right: t = 40.

As seen in Section 5, our algorithms have difficulties in the transitional phase of the double-well system. This is the situation where the variational Gaussian process approximation cannot capture the multi-modal structure of the true posterior process. As we know, the performance of independence samplers is strongly dependent of how well the proposal distribution matches the true posterior. We stress that the intention of the variational MCMC methods we develop in this paper is to address the situation where the state is relatively well observed such that the posterior distribution is essentially uni-modal, a situation one might typically expect in a variety of situations, and in particular in the data assimilation [2] context that motivated this work initially. In any case all naive MCMC approaches, including HMC, will have mixing problems for multi-modal posterior distributions unless specifically adapted [24]

We believe these methods can be extended to cope with larger systems exploiting the variational approximation and could provide a framework for MCMC based inference in more complex, larger stochastic dynamic systems, where methods such as HMC become computationally prohibitive. In future work we plan to assess the ability of sub-optimal variational approximations to provide computationally efficient mechanisms for generating proposal distributions for blocked independence samplers, where we employ localisation in time and space to reduce the computational burden of sampling paths in very high dimensional spaces. We are also considering whether it might make sense to combine the variational sampler with the HMC sampler to address the issue of the poor performance in regions of posterior multi-modality, however this raises so complex questions about combining such sampling methods and maintaining detailed balance in the sampler overall.

References

[1] J. Honerkamp. *Stochastic Dynamical Systems*. VCH Publishers Inc., 1994.

[2] E. Kalnay. *Atmospheric Modelling, Data Assimilation and Predictability*. Cambridge University Press, Cambridge, 2003.

[3] H. J. Kushner. Dynamical equations for optimal filtering. *J. Diff. Eq.*, 3:179–190, 1967.

[4] G. Kitagawa. Non-Gaussian state space modelling of non-stationary time series. *J. Am. Statist. Assoc.*, 82:503–514, 1987.

[5] R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory. *J. Basic Eng. D*, 83:95–108, 1961.

[6] G. Evensen. Sequential data assimilation with a non-linear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.*, 99:10143–10162, 1994.

[7] H. .F Durrant-Whyte S. J. Julier, J. Uhlmann. A new method for the nonlinear tranformation of means and covariances in filters and estimators. *IEEE Trans. on Automatic Control*, 45:477–482, 2000.

[8] C. Archambeau, D. Cornford, M. Opper, and J. Shawe-Tayler. Gaussian process approximations of stochastic differential equations. *Journal of Machine Learning Research*, 1:1–16, 2007.

[9] G. L. Eyink, J. M. Restrepo, and F. J. Alexander. A mean-field approximation in data assimilation for nonlinear dynamics. *Physica D*, 194:347–368, 2004.

[10] C. Andrieu, N. De Freitas, A. Doucet, and M. I. Jordan. An introduction to MCMC for machine learning. *Machine Learning*, 50:5–43, 2003.

[11] R. H. Shumway and D. S. Stoffer. *Time Series Analysis and Its Applications*. Springer, New York, 2000.

[12] F. J. Alexander, G. L. Eyink, and J. M. Restrepo. Accelerated Monte Carlo for optimal estimation of time series. *Journal of Statistical Physics*, 119:1331–1345, 2005.

[13] A. M. Stuart, J. Voss, and P. Winberg. Conditional path sampling of SDEs and the Langevin MCMC method. *Comm. Math. Sci.*, 2:685–697, 2004.

[14] A. Golightly and G. J. Wilkinson. Bayesian sequential inference for nonlinear multivariate diffusions. *Stat Comput*, 16:323–338, 2006.

[15] G. B. Durham and A. R. Gallant. Numerical techniques for maximum likelihood estimation of continuous-time diffusion process. *J. Bus. Econom. Statist.*, 20:297–338, 2002.

[16] G. O. Roberts A. Beskos, O. Papaspiliopoulous and P. Fearnhead. Exact and computationally efficient likelihood-based estimation for discretely observed diffusion processes. *J. R. Statist. Soc. B*, 68:333–382, 2006.

[17] Nando de Freitas, Pedro Hϕjen-Sϕrensen, Michael Jordan, and Stuart Russell. Variational MCMC. In *Proceedings of the 17th Annual Conference on Uncertainty in Artificial Intelligence*, pages 120–127, 2001.

[18] S. C. Zhu, R. Zhang, and Z. W. Tu. Integrating top-down/bottom-up for objection recognition by data-driven Markov Chain Monte Carlo. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition 2000*, 120–127, 2000.

[19] P. E. Klöden and Eckhard Platen. *Numerical Solution of Stochastic Differential Equations*. Spinger-Verlag, Berlin, 1992.

[20] P. Del Moral D. Crisan and T. J. Lyons. Interacting particle systems approximations of the Kushner-Stratonovich equation. *Advances in Applied Probability*, 31:819–838, 1999.

[21] C. Archambeau, M. Opper, Y. Shen, D. Cornford, and J. Shawe-Tayler. Variational Inference for Diffusion Processes. In *Advances In Neural Information Processing Systems 20*, The MIT Press, Cambridge, Massachusetts. Accepted (to appear)

[22] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Bometrika*, 57:97–109, 1970.

[23] R.N. Miller, E. F. Carter, and S. T. Blue. Data assimilation into nonlinear stochastic models. *Tellus A*, 51:167–194, 1999.

[24] Cornford, D., Csato, L., Evans, D. J. and Opper, M. Bayesian analysis of the scatterometer wind retrieval inverse problem: some new approaches, *Journal of the Royal Statistical Society - B*, pages 609–626, 2004.

A Kernel Test of Nonlinear Granger Causality

Xiaohai Sun
Max Planck Institute for Biological Cybernetics
72076 Tübingen, Germany
xiaohai.sun@tuebingen.mpg.de

Abstract

We present a novel test of nonlinear Granger causality in bivariate time series. The trace norm of conditional covariance operators is used to capture the prediction errors. Based on this measure, a subsampling-based multiple testing procedure tests the prediction improvement of one time series by the other one. The distributional properties of the resulting p-values reveal the direction of Granger causality. Encouraging results of experiments with simulated and real-world data support our approach.

1 Introduction

In this paper, a time series $\mathrm{X} := (\ldots, x_{t-1}, x_t, x_{t+1}, \ldots)^{\mathrm{T}}$ is a discrete time, continuous state process where $t \in \mathbb{Z}$ is a certain discrete time point. Time points are usually taken at equally spaced intervals. Given a bivariate time series (X, Y) measured simultaneously, we focus on the problem whether the underlying process of X is causal to the underlying process of Y and/or the other way around. The well-known concept of causality in analysis of times series is the so-called Granger causality: The process X Granger causes another process Y, subsequently denoted as "$\mathrm{X} \Rightarrow \mathrm{Y}$", if future values of Y can be better predicted using the past values of (X, Y) compared to using the past values of Y alone. To formalize the time flow, we introduce the notation of the time-delayed embedding vector reconstructing the state (or phase) space of times series X, which is expressed as $\mathrm{X}_t^{n,r} := (x_{t-(n-1)r}, \ldots, x_{t-2r}, x_{t-r}, x_t)^{\mathrm{T}}$, where n is the embedding dimension and r is the time delay (or lag) between successive elements of the state vector [1, 2]. The choice of r, n depends on the dynamics of underlying process. We refer to [3, 4] for some principled way of choosing r, n. Throughout this paper, we set $r = 1$ and the expression $\mathrm{X}_t := (x_{t-n+1}, \ldots, x_{t-1}, x_t)^{\mathrm{T}}$ is used for $n > 1$. For $n = 1$, we use the notation x_t explicitly, in stead of X_t.

Fig. 1 illustrates the task of inferring Granger causality from X to Y in terms of embedding vectors $\ldots, \mathrm{X}_t, \mathrm{X}_{t+1}, \ldots$ and $\ldots, \mathrm{Y}_t, \mathrm{Y}_{t+1}, \ldots$. We use the notation "$\mathrm{X}_t \rightarrow \mathrm{Y}_{t+1} \,|\, \mathrm{Y}_t, \mathrm{Y}_{t-1}, \ldots$" to describe the conditional predictability of Y_{t+1} by X_t given the past observations $(\mathrm{Y}_t, \mathrm{Y}_{t-1}, \ldots$ of $\mathrm{Y}_{t+1})$. Note that, we distinguish between simple arrow "$\rightarrow$" expressing predictability between time slides and the double arrow "$\Rightarrow$" expressing Granger causality between time series.

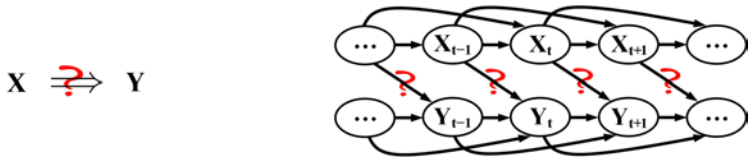


Figure 1: Inferring Granger causality from X to Y (left plot) can be expressed in terms of testing the prediction improvement of Y_{t+1} by X_t (right plot).

To assess the predictability "$\mathrm{X}_t \rightarrow \mathrm{Y}_{t+1} \,|\, \mathrm{Y}_t, \mathrm{Y}_{t-1}, \ldots$", the standard test of Granger causality developed by Granger [5] considers the following autoregressive models:

$$y_{t+1} = \alpha^{\mathrm{T}} \cdot \mathrm{Y}_t + \xi^{(\mathrm{Y})} \quad \text{and} \quad y_{t+1} = a^{\mathrm{T}} \cdot \mathrm{Y}_t + b^{\mathrm{T}} \cdot \mathrm{X}_t + \xi^{(\mathrm{Y}|\mathrm{X})} \,,$$

where $\xi^{(\mathrm{Y})}$ and $\xi^{(\mathrm{Y}|\mathrm{X})}$ represent the prediction errors, $\alpha = (\alpha_1, \ldots, \alpha_n)^{\mathrm{T}}$, $a = (a_1, \ldots, a_n)^{\mathrm{T}}$, $b = (b_1, \ldots, b_n)^{\mathrm{T}}$ denote regression coefficient vectors. The coefficient vectors are determined so that the variances of $\xi^{(\mathrm{Y})}$ and $\xi^{(\mathrm{Y}|\mathrm{X})}$ minimize. Once the coefficient vectors have been calculated, the causal inference of X on Y can be revealed if the variance $\mathrm{Var}[\xi^{(\mathrm{Y}|\mathrm{X})}]$ is significantly smaller than $\mathrm{Var}[\xi^{(\mathrm{Y})}]$, which means that the value of X additionally improves the prediction of the future value of Y, after the past observations of Y have been considered. The opposite direction "$\mathrm{X} \Leftarrow \mathrm{Y}$" can be tested analogously. Such traditional test of Granger causality is based on linear regression models, and its application to nonlinear systems may not be appropriate in the general case. A nonlinear extension of Granger causality, called the extended Granger causality index was proposed in [6]. The main idea of this technique is to divide the phase space into a set of small neighborhoods and approximate the globally nonlinear dynamics by local linear regression models. Obviously, the local linearity is a restrictive assumption. Another recently introduced nonlinear extensions of Granger causality [7] completely dropped the linearity assumption and based the prediction on kernel autoregression scheme

$$y_{t+1} = \sum_i \alpha_i \cdot \phi_i(\mathrm{Y}_t) + \xi^{(\mathrm{Y})} \quad \text{and} \quad y_{t+1} = \sum_i a_i \cdot \phi_i(\mathrm{Y}_t) + \sum_j b_j \cdot \psi_j(\mathrm{X}_t) + \xi^{(\mathrm{Y}|\mathrm{X})}$$

with regression coefficients α_i, a_i, b_j and some nonlinear functions ϕ_i, ψ_j, e.g., nonlinear radial based functions (RBFs). Nonetheless, this approach assumes additive interactions between $\phi_i(Y_t)$ and $\psi_j(\mathrm{X}_t)$. In this paper, we present an autoregression model, in which both linearity and additivity assumptions are dropped, as follows:

$$\begin{aligned} \sum_i \alpha_i \cdot \phi_i(\mathrm{Y}_{t+1}) &= \sum_j \beta_j \cdot \psi_j(\mathrm{Y}_t, \ldots, \mathrm{Y}_{t-l+1}) + \xi^{(\mathrm{Y})} \quad \text{and} \\ \sum_i a_i \cdot \phi_i(\mathrm{Y}_{t+1}) &= \sum_j b_j \cdot \psi_j(\mathrm{Y}_t, \ldots, \mathrm{Y}_{t-l+1}, \mathrm{X}_t) + \xi^{(\mathrm{Y}|\mathrm{X})} \end{aligned} \tag{1}$$

with regression coefficients $\alpha_i, \beta_j, a_i, b_j$, some positive integer l, and some nonlinear functions ϕ_i, ψ_j. In addition, the target variable Y_{t+1} in our model is also represented by nonlinear functions. We will choose the nonlinear functions ϕ_i, ψ_j from the so-called reproducing kernel Hilbert space [8] (RKHS). For this purpose, we introduce the so-called kernel framework.

2 Kernel Framework

A positive definite kernel $k_{\mathcal{X}} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ on a non-empty set $\mathcal{X}$ is a symmetric function, i.e., $k_{\mathcal{X}}(x, x') = k_{\mathcal{X}}(x', x)$ for any $x, x' \in \mathcal{X}$ such that for arbitrary $n \in \mathbb{N}$ and $x^{(1)}, \ldots, x^{(n)} \in \mathcal{X}$ the matrix K with $(K)_{ij} := k_{\mathcal{X}}(x^{(i)}, x^{(j)})$ is positive definite, i.e., $\sum_{i,j=1}^{n} c_i c_j k_{\mathcal{X}}(x^{(i)}, x^{(j)}) \geq 0$ for all $c_1, \ldots, c_n \in \mathbb{R}$. An RKHS $\mathcal{H}_{\mathcal{X}}$ is a Hilbert space defined by the completion of an inner product space of functions $k_{\mathcal{X}}(x, \cdot)$ with $x \in \mathcal{X}$ and the inner product defined by

$$\langle k_{\mathcal{X}}(x, \cdot), k_{\mathcal{X}}(x', \cdot) \rangle = k_{\mathcal{X}}(x, x')$$

for all $x, x' \in \mathcal{X}$. In other words, $\phi(x)(\cdot) = k_{\mathcal{X}}(x, \cdot)$ defines a map from $\mathcal{X}$ into a feature space $\mathcal{H}_{\mathcal{X}}$. With the so-called "kernel trick", a linear algorithm can easily be transformed into a non-linear algorithm, which is equivalent to the linear algorithm operating in the space of ϕ. However, the mapping ϕ is never explicitly computed, since the kernel function is used for calculating the inner product. This is desirable, because the high-dimensional space may be infinite-dimensional, as is the case when the kernel is, e.g., a Gaussian:

$$k_{\mathcal{X}} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}, \quad k_{\mathcal{X}}(x, x') = \exp(-\|x - x'\|^2 / 2\sigma^2) \,. \tag{2}$$

In this framework, we are able to define the covariance operator [9] expressing variance of variables in the feature space.

2.1 Covariance Operator

Suppose we have random vector (X, Y) taking values on $\mathcal{X} \times \mathcal{Y}$. The base spaces $\mathcal{X}$ and $\mathcal{Y}$ are topological spaces. Measurability of these spaces is defined with respect to the Borel σ-field. The

joint distribution of (X, Y) is denoted by P_{XY} and the marginal distributions by P_X and P_Y. Let $(\mathcal{X}, \mathcal{B}_\mathcal{X})$ and $(\mathcal{Y}, \mathcal{B}_\mathcal{Y})$ be measurable spaces and $(\mathcal{H}_\mathcal{X}, k_\mathcal{X})$, $(\mathcal{H}_\mathcal{Y}, k_\mathcal{Y})$ be RKHSs of functions on $\mathcal{X}$ and $\mathcal{Y}$ with positive definite kernels $k_\mathcal{X}, k_\mathcal{Y}$. We consider only random vectors (X, Y) on $\mathcal{X} \times \mathcal{Y}$ such that the expectations $\mathrm{E}_X[k_\mathcal{X}(X, X)], \mathrm{E}_Y[k_\mathcal{Y}(Y, Y)]$ are finite, which guarantees $\mathcal{H}_\mathcal{X}$ and $\mathcal{H}_\mathcal{Y}$ are included in $L^2(P_X)$ and $L^2(P_Y)$ respectively, where $L^2(\mu)$ denotes the Hilbert space of square integrable functions with respect to a measure μ. It is known that there exists a unique operator Σ_{YX}, called cross-covariance operator, from $\mathcal{H}_\mathcal{X}$ to $\mathcal{H}_\mathcal{Y}$ such that

$$\langle g, \Sigma_{YX} f \rangle_{\mathcal{H}_\mathcal{Y}} = \mathrm{E}_{XY}[f(X)g(Y)] - \mathrm{E}_X[f(X)]\mathrm{E}_Y[g(Y)] = \mathrm{Cov}[f(X), g(Y)]$$

for all $f \in \mathcal{H}_\mathcal{X}, g \in \mathcal{H}_\mathcal{Y}$. Here, $\mathrm{Cov}[\cdot]$ denotes the covariance. $\mathrm{E}_X[\cdot]$, $\mathrm{E}_Y[\cdot]$ and $\mathrm{E}_{XY}[\cdot]$ denote the expectation over P_X, P_Y and P_{XY}, respectively. Baker [9] showed that Σ_{YX} has a representation of the form $\Sigma_{YX} = \Sigma_{YY}^{1/2} V_{YX} \Sigma_{XX}^{1/2}$ with a unique bounded operator $V_{YX} : \mathcal{H}_\mathcal{X} \to \mathcal{H}_\mathcal{Y}$ such that $\|V_{YX}\| \leq 1$, where $\|\cdot\|$ is used for the operator norm of a bounded operator, i.e., $\|V\| = \sup_{\|f\|=1} \|Vf\|$. Moreover, it is obvious that $\Sigma_{XY} = \Sigma_{YX}^*$, where Σ^* denotes the adjoint of an operator Σ. If X is equal to Y, the positive self-adjoint operator Σ_{YY} is the covariance operator.

Based on the cross-covariance operator, we introduce the conditional covariance operator. Let $(\mathcal{H}_\mathcal{X}, k_\mathcal{X})$, $(\mathcal{H}_\mathcal{Y}, k_\mathcal{Y})$ be RKHSs on measurable spaces $\mathcal{X}, \mathcal{Y}$ respectively. Let (X, Y) be a random vector on $\mathcal{X} \times \mathcal{Y}$. The positive self-adjoint operator

$$\Sigma_{YY|X} := \Sigma_{YY} - \Sigma_{YY}^{1/2} V_{YX} V_{XY} \Sigma_{YY}^{1/2}$$

is called the conditional covariance operator, where V_{YX} and V_{XY} are the bounded operators derived from Σ_{YX} and Σ_{XY}. If Σ_{XX}^{-1} exists, we can rewrite $\Sigma_{YY|X}$ as

$$\Sigma_{YY|X} = \Sigma_{YY} - \Sigma_{YX} \Sigma_{XX}^{-1} \Sigma_{XY} .$$

Fukumizu et al. [10, Proposition 2] showed that

$$\langle g, \Sigma_{YY|X} g \rangle_{\mathcal{H}_\mathcal{Y}} = \inf_{f \in \mathcal{H}_\mathcal{X}} \mathrm{E}_{XY}\left[\left| (g(Y) - \mathrm{E}_Y[g(Y)]) - (f(X) - \mathrm{E}_X[f(X)]) \right|^2 \right]$$

for any $g \in \mathcal{H}_\mathcal{Y}$. This is an analogous to the well-known results on covariance matrices and linear regression: The conditional covariance matrix $C_{YY|X} = C_{YY} - C_{YX} C_{XX}^{-1} C_{XY}$ expresses the residual error of the least square regression problem as $b^{\mathrm{T}} C_{YY|X} b = \min_a \mathrm{E}_{XY} \|b^{\mathrm{T}} Y - a^{\mathrm{T}} X\|^2$. To relate this residual error to the conditional variance of $g(Y)$ given X, the following assumption for RKHSs is made.

Assumption 1 *Let* $\mathbf{1}$ *denote the function with constant value* 1 *on* $\mathcal{X}$. *Then* $\mathcal{H}_\mathcal{X} + \mathbb{R} \cdot \mathbf{1}$ *is dense in* $L^2(P_X)$, *where "*$+$*" means the sum of Hilbert spaces.*

The kernels that satisfy this assumption are necessarily "characteristic". The notation of the characteristic kernels is a generalization of the characteristic function $\mathrm{E}_X[\exp(\sqrt{-1} u^{\mathrm{T}} X)]$, which is the expectation of the (complex-valued) positive definite kernel $k(x, u) = \exp(\sqrt{-1} u^{\mathrm{T}} x)$ (see [11, 12] for more details). One popular class of characteristic kernels is the universal kernels [13] on a compact metric space, e.g., the Gaussian or Laplacian kernel on a compact subset of $\mathbb{R}^m$, because the Banach space of bounded continuous functions on a compact subset $\mathcal{X}$ of $\mathbb{R}^m$ is dense in $L^2(P_X)$ for any P_X on $\mathcal{X}$. Another example is the Gaussian or Laplacian kernel on the entire Euclidean space, since many random variables are defined on non-compact spaces. One can prove that, Assumption 1 holds for these kernels (see Appendix in [11]). A recent paper of Sriperumbudur et al. [12] showed the necessary and sufficient condition, under which shift-invariant kernels are characteristic. Shift-invariant kernels are given by $k(x, x') = \psi(x - x')$ where ψ is a bounded continuous real-valued positive definite function on $\mathbb{R}^m$, e.g., Gaussian, Laplacian, etc.

Under Assumption 1, one can show that

$$\langle g, \Sigma_{YY|X} g \rangle_{\mathcal{H}_\mathcal{Y}} = \mathrm{E}_X\left[\mathrm{Var}_{Y|X}[g(Y)|X] \right]$$

for all $g \in \mathcal{H}_\mathcal{Y}$. Thus, the conditional covariance operator expresses the conditional variance of $g(Y)$ given X in the feature space. As a side note, Ancona et al. [14] claimed that not all kernels are suitable for their nonlinear prediction schemes. They presented sufficient conditions, which hold for Gaussian kernels. Our kernel framework allows wider class of kernels. Note that the maps

ϕ, ψ in Eq. 1, do not necessarily belong to the same function class, even though we overall used Gaussian kernels in our experiments. The parameter σ^2 in kernel functions as in Eq. 2 is adapted to the variance of variables independently. In our experiments, we set the parameter such that $2\sigma^2$ equals the variance of the corresponding variable.

To evaluate the conditional covariance operator, we use the trace norm, because it is not difficult to see that the trace norm of the operator is directly linked with the sum of residual errors, namely

$$\mathrm{Tr}(\Sigma_{YY|X}) = \sum_i \min_{f \in \mathcal{H}_\mathcal{X}} \mathrm{E}_{XY}\left[\left|(\phi_i(Y) - \mathrm{E}_Y[\phi_i(Y)]) - (f(X) - \mathrm{E}_X[f(X)])\right|^2\right],$$

where $\{\phi_i\}_{i=1}^\infty$ is the complete orthonormal system of the separable RKHS $\mathcal{H}_\mathcal{Y}$. An RKHS $(\mathcal{H}_\mathcal{Y}, k_\mathcal{Y})$ is separable, when the topological space $\mathcal{Y}$ is separable and $k_\mathcal{Y}$ is continuous on $\mathcal{Y}\times\mathcal{Y}$ [15].

Further, let $(\mathcal{H}_{\mathcal{X}_1}, k_{\mathcal{X}_1})$, $(\mathcal{H}_{\mathcal{X}_2}, k_{\mathcal{X}_2})$ be RKHSs on measurable spaces $\mathcal{X}_1, \mathcal{X}_2$ respectively. If we define $X := (X_1, X_2)$ and $\mathcal{H}_\mathcal{X} := \mathcal{H}_{\mathcal{X}_1} \otimes \mathcal{H}_{\mathcal{X}_2}$, it can be shown that $\Sigma_{YY|X} \leq \Sigma_{YY|X_1}$, where the inequality refers to the usual order of self-adjoint operators, namely if $A \leq B \Leftrightarrow \langle Ag, g\rangle \leq \langle Bg, g\rangle$ for all $g \in \mathcal{H}_\mathcal{Y}$. Further, if $\mathcal{H}_\mathcal{X}$, $\mathcal{H}_\mathcal{Y}$, $\mathcal{H}_{\mathcal{X}_1}$ are given by characteristic kernels, $\Sigma_{YY|X_1,X_2} = \Sigma_{YY|X_1} \Leftrightarrow Y \perp\!\!\!\perp X_2 \,|\, X_1$, which denotes that Y and X_2 are conditionally independent, given X_1. In terms of the trace norm, we have the following property:

Property 1 *Let $\mathbb{T}_{YY|X}$ denote the trace norm of the conditional covariance operator $\mathrm{Tr}(\Sigma_{YY|X})$ with $X := (X_1, X_2)$. Then we have*

$$\mathbb{T}_{YY|X_1,X_2} < \mathbb{T}_{YY|X_1} \;\Leftrightarrow\; Y \not\perp\!\!\!\perp X_2 \,|\, X_1 \quad \textit{and} \quad \mathbb{T}_{YY|X_1,X_2} = \mathbb{T}_{YY|X_1} \;\Leftrightarrow\; Y \perp\!\!\!\perp X_2 \,|\, X_1\,.$$

Property 1 of the trace norm generalizes the (P1)-property, required by Ancona et al. [7, Section II.A] for any measure of nonlinear Granger causality, since (P1)-property describes merely the bivariate case, while Property 1 holds also for multivariate cases.

2.2 Empirical Estimation of Operators

In analogy to the work of [11], we introduce the estimations of $\mathbb{T}_{YY}$ and $\mathbb{T}_{YY|X}$ based on sample $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})$ from the joint distribution. Using the empirical mean elements $\widehat{m}_X^{(n)} = \frac{1}{n}\sum_{i=1}^n k_\mathcal{X}(x^{(i)}, \cdot)$ and $\widehat{m}_Y^{(n)} = \frac{1}{n}\sum_{i=1}^n k_\mathcal{Y}(y^{(i)}, \cdot)$, an estimator of Σ_{YX} is

$$\widehat{\Sigma}_{YX}^{(n)} = \frac{1}{n}\sum_{i=1}^n \left(k_\mathcal{Y}(y^{(i)}, \cdot) - \widehat{m}_Y^{(n)}\right)\langle k_\mathcal{X}(x^{(i)}, \cdot) - \widehat{m}_X^{(n)}, \cdot\rangle_{\mathcal{H}_\mathcal{X}}\,.$$

$\widehat{\Sigma}_{YY}^{(n)}$ and $\widehat{\Sigma}_{XX}^{(n)}$ can be defined accordingly. An estimator of $\Sigma_{YY|X}$ is then defined by

$$\widehat{\Sigma}_{YY|X}^{(n,\epsilon)} = \widehat{\Sigma}_{YY}^{(n)} - \widehat{\Sigma}_{YX}^{(n)}(\widehat{\Sigma}_{XX}^{(n)} + \epsilon I)^{-1}\widehat{\Sigma}_{XY}^{(n)}\,,$$

where $\epsilon > 0$ is a regularization constant that enables inversion.[1] It can be shown that $\widehat{\mathbb{T}}_{YY}^{(n)} = \mathrm{Tr}(\widehat{\Sigma}_{YY}^{(n)})$ is a consistent estimator of $\mathbb{T}_{YY}$, which guarantees to converge in Hilbert-Schmidt norm at rate $n^{-1/2}$. Moreover, $\widehat{\mathbb{T}}_{YY|X}^{(n,\epsilon)} = \mathrm{Tr}(\widehat{\Sigma}_{YY|X}^{(n,\epsilon)})$ is a consistent estimator of $\mathbb{T}_{YY|X}$. If ϵ converges to zero more slowly than $n^{-1/2}$, this estimator converges to $\mathbb{T}_{YY|Z}$. For notational convenience, we will henceforth omit the upper index and use $\widehat{\mathbb{T}}_{YY}$ and $\widehat{\mathbb{T}}_{YY|X}$ to denote the empirical estimators.

The computation with kernel matrices of n data points becomes infeasible for very large n. In our practical implementation, we use the incomplete Cholesky decomposition $\widehat{K} = LL^\mathrm{T}$ [18] where L is a lower triangular matrix determined uniquely by this equation. This may lead to considerably fewer columns than the original matrix. If k columns are returned, the storage requirements are $O(kn)$ instead of $O(n^2)$, and the running time of many matrix operations reduces from $O(n^3)$ to $O(nk^2)$.

[1] The regularizer is required as the number of observed data points is finite, whereas the feature space could be infinite-dimensional. The regularization may be understood as a smoothness assumption on the eigenfunctions of $\mathcal{H}_\mathcal{X}$. It is analogous to Tikhonov regularization [16] or ridge regression [17]. Many simulated experiments showed that the empirical measures are insensitive to ϵ, if it is chosen in some appropriate interval, e.g., $[10^{-10}, 10^{-2}]$. We chose $\epsilon = 10^{-5}$ in all our experiments.

3 Subsampling-based Testing of Granger Causality

We have showed that $\widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1}}$ and $\widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t}$ can be respectively considered as measures for variances of prediction errors $\xi^{(\mathrm{Y})}$ and $\xi^{(\mathrm{Y}|\mathrm{X})}$ based on models defined in Eq. 1. To test the significance of the relation between them in spite of statistical fluctuations, we employ the permutation test. For technical reasons, we first rephrase the autoregression models of Eq. 1 for some integer $l>0$ as follows

$$\begin{aligned}\sum_{i=1}^{n}\alpha_i\cdot\phi_i(\mathrm{Y}_{t+1}) &= \sum_{j=1}^{n}\beta_j\cdot\psi_j(\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t^{\pi})+\xi^{(\mathrm{Y}|\mathrm{X}^{\pi})} \quad \text{and}\\ \sum_{i=1}^{n}a_i\cdot\phi_i(\mathrm{Y}_{t+1}) &= \sum_{j=1}^{n}b_j\cdot\psi_j(\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t)+\xi^{(\mathrm{Y}|\mathrm{X})}\end{aligned} \tag{3}$$

where $\mathrm{X}_t^{\pi}:=(x_{\pi(t-n+1)},\ldots,x_{\pi(t)})^{\mathrm{T}}$ denotes the data vector obtained by shuffling n data points of X_t by a random permutation π. For the special case $l=0$, the models of Eq. 3 are defined by

$$\begin{aligned}\sum_{i=1}^{n}\alpha_i\cdot\phi_i(\mathrm{Y}_{t+1}) &= \sum_{j=1}^{n}\beta_j\cdot\psi_j(\mathrm{X}_t^{\pi})+\xi^{(\mathrm{Y}|\mathrm{X}^{\pi})} \quad \text{and}\\ \sum_{i=1}^{n}a_i\cdot\phi_i(\mathrm{Y}_{t+1}) &= \sum_{j=1}^{n}b_j\cdot\psi_j(\mathrm{X}_t)+\xi^{(\mathrm{Y}|\mathrm{X})}\,.\end{aligned} \tag{4}$$

Then, we choose ϕ_i as nonlinear maps of $\mathcal{Y}$ (set of all possible values of Y_{t+1}) into feature space $\mathcal{H}_{\mathcal{Y}}$, and ψ_j as nonlinear maps of $\mathcal{Y}^l\times\mathcal{X}$ (set of all possible values of $(\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t)$) into feature space $\mathcal{H}_{\mathcal{Y}^l\times\mathcal{X}}$. In practice, the feature space $\mathcal{H}_{\mathcal{Y}^l\times\mathcal{X}}$ is spanned by data vector $(\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t^{\pi})$ or by data vector $(\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t)$. To make both spaces coincide, we restrict the random permutation π to those that satisfy the condition

$$(\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t^{\pi})\equiv(\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t)\,. \tag{5}$$

If vector X_t takes only discrete/categorical values, the condition of Eq. 5 restricts π to permutations within the same category. In the case of real-valued X_t, Eq. 5 could be said to hold if X_t^{π} and X_t are "similar" in some sense. This suggests the use of clustering techniques to search for an appropriate partition of data points of X_t. In our experiments, we applied the standard k-means clustering algorithm. Other clustering algorithms can be applied as well. Using a set of random permutations $\pi=\{\pi_1,\ldots,\pi_k\}$ satisfying Eq. 5, the null distribution (under unpredictability) of $\mathrm{Var}[\xi^{(\mathrm{Y})}]$ can be simulated. Based on the empirical null distribution

$$\{\widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t^{\pi_1}},\ldots,\widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t^{\pi_k}}\}\,, \tag{6}$$

p-value can be determined, which is the percentage of values in the set of Eq. 6, which are properly less than $\widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t}$. Smaller p-values suggest stronger evidence against the null hypothesis (unpredictable), and thus stronger evidence favoring the alternative hypothesis (predictable). As a cut-off point for p-values, a significance level α is pre-specified. A typical choice is $\alpha=0.05$. If $p<\alpha$, the null hypothesis is rejected, which means the predictability "$\mathrm{X}_t\rightarrow\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1}$" is significantly verified; Otherwise, we accept "$\mathrm{X}_t\not\rightarrow\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1}$". In summary, the p-value expressing the relationship between $\mathrm{Var}[\xi^{(\mathrm{Y}|\mathrm{X})}]$ and $\mathrm{Var}[\xi^{(\mathrm{Y}|\mathrm{X}^{\pi})}]$ gives the evidence for the Granger causality "$\mathrm{X}\Rightarrow\mathrm{Y}$".

The remaining problem is the choice of l in Eq. 3, which states that $(\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1})$ achieves the maximum knowledge that would be useful to predict Y_{t+1}. We propose to calculate l by a iterative procedure based on following autoregression models

$$\begin{aligned}\sum_{i=1}^{n}\alpha_i\cdot\phi_i(\mathrm{Y}_{t+1}) &= \sum_{j=1}^{n}\beta_j\cdot\psi_j(\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{Y}_{t-l}^{\pi})+\xi_l^{(\pi)} \quad \text{and}\\ \sum_{i=1}^{n}a_i\cdot\phi_i(\mathrm{Y}_{t+1}) &= \sum_{j=1}^{n}b_j\cdot\psi_j(\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{Y}_{t-l})+\xi_l\,.\end{aligned} \tag{7}$$

The integer $l \in \{0, 1, \ldots\}$ in Eq. 3 is specified to be the smallest integer where $\mathrm{Var}[\xi_l]$ is not significantly less than $\mathrm{Var}[\xi_l^{(\pi)}]$. In other words, l is specified to be the smallest nonnegative integer where "$\mathrm{Y}_{t-l} \not\rightarrow \mathrm{Y}_{t+1} | \mathrm{Y}_t, \ldots, \mathrm{Y}_{t-l+1}$". As mentioned previously, the models in Eq. 4 are used, if $l = 0$.

So far, we have shown the test of predictability of Y_{t+1} by X_t on a single sample. The statistical power of such single tests is often limited, since many real-world time series are nonstationary, in particular, the causal relationship could vary over time. For this reason, we propose a subsampling-based multiple testing procedure and utilize the distributional properties of the resulting p-values based on different sub-time-series (embedding vectors). Fig. 2 summarizes our multiple testing procedure for detecting Granger causality. Step 1 runs single tests on N random sub-time-series and obtains N p-values, one for each sub-time-series.

Input: Bivariate time series (X, Y) and hypothesis $\mathrm{X} \Rightarrow \mathrm{Y}$.

Step 1: For each sub-time-series $(\mathrm{X}_{t_i}, \mathrm{Y}_{t_i})$, $i = 1, \ldots, N$:

Step 1.1: Determine the smallest nonnegative integer l such that $\mathrm{Y}_{t_i - l} \not\rightarrow \mathrm{Y}_{t_i+1} | \mathrm{Y}_{t_i}, \ldots, \mathrm{Y}_{t_i-l+1}$.

Step 1.2: Calculate the p-value p_i based on the models in Eq. 4 ($l = 0$) or the models in Eq. 3 ($l > 0$).

Step 2: Calculate the skewness of $p_1, \ldots, p_N$ and the probability P^{Unif} that $p_1, \ldots, p_N$ are uniformly distributed over $[0, 1]$.

Output: If $P^{\text{Unif}} < \frac{1}{2}$ and p-values are positively skewed, accept the hypothesis; Otherwise, reject it.

Figure 2: Subsampling-based multiple testing of Granger causality

Based on the distributional properties of these p-values (Step 2), the procedure makes the decision on the predictability. To make this step apparent, we take a closer look at the distribution of p-values. According to Property 1, the relation

$$\widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t} \leq \widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1}} \,,$$

holds in general. And, if the prediction of Y_{t+1} can indeed be improved by X_t, we will expect that

$$\widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t} < \widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t^\pi} \,. \tag{8}$$

Roughly speaking, the prediction improvement of Y_{t+1} by X_t is reflected in a significant reduction of the sum of residual errors, captured by the relation between $\widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t}$ and $\widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t^\pi}$. Thus, the majority of p-values are closer to 0 and the distribution of p-values is positively skewed (right-skewed). If X_t does not improve the prediction of Y_{t+1},

$$\widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t} \geq \widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t^\pi} \tag{9}$$

is as likely as the relation described in Eq. 8. Consequently, p-values are uniformly distributed over $[0, 1]$ and the skewness of p-values vanishes. If the relation in Eq. 9 is true for the majority of random permutations, more p-values are closer to 1 and the distribution of p-values is negatively skewed (left-skewed). This case, called uncertain situation, can occur due to various reasons. One imaginable situation is, e.g., $\widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t} = \widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t^\pi}$, the null distribution is degenerate (all its probability mass is concentrated on one point). For instance, if $(\mathrm{Y}_t, \ldots, \mathrm{Y}_{t-l+1})$ is high-dimensional, i.e., l is large, or the statistical fluctuation of Y_{t+1} is very small, it could occur that

$$\widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t} \approx \widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1},\mathrm{X}_t^\pi} \approx \widehat{\mathbb{T}}_{\mathrm{Y}_{t+1}\mathrm{Y}_{t+1}|\mathrm{Y}_t,\ldots,\mathrm{Y}_{t-l+1}} \approx 0 \,.$$

Then, the relation between these measures cannot provide any reliable information about the predictability of Y_{t+1} by X_t. We interpret such uncertain situations as no evidence for the Granger causality hypothesis "$\mathrm{X} \Rightarrow \mathrm{Y}$".

Inspired by [19, 20], we visualize these observations by an intuitive graphical tool. We first sort the set of p-values $\{p_{(1)}, \ldots, p_{(N)}\}$ in an increasing order, i.e., $p_1 \leq p_2 \leq \ldots \leq p_N$. If p_i behaves as an ordered sample from the uniform distribution over $[0, 1]$, the expected value of p_i is approximately $\frac{i}{N}$. The slope of p_i versus i, also called Q-Q plot ("Q" stands for quantile), should exhibit a linear relationship, along a line of slope $\frac{1}{N}$ passing through the origin (diagonal line in the Q-Q plot as

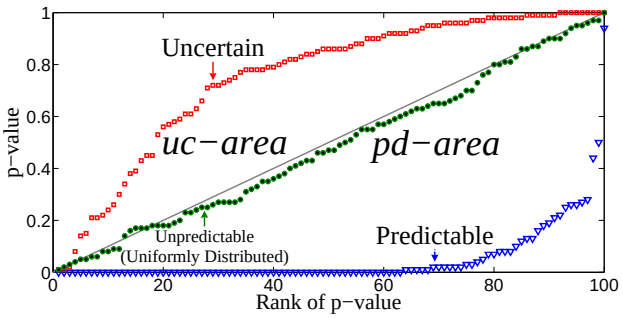


Figure 3: Q-Q plots of three sets of p-values obtained by three multiple tests. The field above and under the diagonal line are called uncertain (uc) area and predictable (pd) area respectively.

shown in Fig. 3). If p-values are positively skewed, the reordered p-values are located in the subfield under the diagonal line, called pd-area ("pd": predictable). If p-values are negatively skewed, the reordered p-values are located in the so-called uc-area ("uc": uncertain).

For a reliable decision based on the resulting p-values, the subsamples should be independent to some degree, since independent statistics on p-values are needed. This is the case, when the given sample size is much larger than the subsample size (dimension n of the embedding vector). In our experiments, we fixed the subsample size at 100, since time series in our experiments contain at least 5000 data points. The other parameter of the multiple testing is the number of replications of the single test: N. In principle, N should be large enough to enable a reliable identification of uniform distribution from N p-values. In our experiments, we chose $N \geq 100$. For large sample sizes, we chose $N = 1000$.

The last question is how to judge, in spite of the fluctuation of one specific set of p-values, whether the N resulting p-values are uniformly distributed or not. We transform this problem to a two-sample-problem. More precisely, we simulate 1000 samples of N values from the uniform distribution over $[0, 1]$. For each of the 1000 simulated samples, we test whether the N resulting p-values are identically distributed with the N values from truly uniform distribution. The percentage of the positive results, i.e., the resulting p-values and the simulated values come from the same distribution, can be considered as the probability that the resulting p-values are uniformly distributed: P^{Unif}. If $P^{\text{Unif}} < \frac{1}{2}$, the p-values are less likely from a uniform distribution than from a non-uniform distribution. In our experiments, we employ the kernel-based test for the two-sample-problem proposed by Gretton et al. [21]. After all, the decision of Granger causality relies on whether $P^{\text{Unif}} < \frac{1}{2}$ and whether the p-values are positively skewed.

4 Experiments

To demonstrate the effectiveness of the proposed approach, we test our algorithm on simulated data generated by chaotic maps and real-life systems of different scientific fields: financial time series and a physiological problem.

4.1 Hénon Maps

As the first simulated example, we consider the following two noisy Hénon maps:

$$\begin{aligned} x_{t+1} &= a + c_1\, x_{t-1} - d_1\, x_t^2 + \mu\, \xi_1 \\ y_{t+1} &= a + c_2\, y_{t-1} - b\, x_t y_t - (1-b)\, d_2\, y_t^2 + \mu\, \xi_2 \end{aligned}$$

represented as systems X and Y, respectively. Here, system X drives system Y with coupling strength $b \in [0, 1]$. If $b = 0$, X and Y are uncoupled; If $b > 0$, we have a uni-directed coupling $\mathrm{X} \Rightarrow \mathrm{Y}$. This example is also studied by Bhattacharya et al. [22], who proposed to choose $b < 0.7$ to avoid strong synchronization. Similar to [22], we fixed the parameters at $a = 1.4$, $c_1 = 0.3$, $c_2 = 0.1$, $d_1 = 1$, $d_2 = 0.4$, $\mu = 0.01$. ξ_1, ξ_2 are unit variance Gaussian distributed noise terms. Note that X and Y are different systems even in the case of $b = 0$, because $c_1 \neq c_2$ and $d_1 \neq d_2$. Therefore, identical

synchronization is impossible. We start with points $(x_1, y_1) = (x_2, y_2) = (0, 0)$. The first and the third plot (from left) of Fig. 4 show the time series of 10000 data points. We ran our test procedure on uncoupled time series ($b = 0$) and weakly unidirectionally coupled time series ($b = 0.25$). The reordered p-values obtained in both cases are visualized in the second and fourth plot of Fig. 4. In the case of $b = 0$, our test rejected the Granger causality in both directions. In the case of $b = 0.25$, our test revealed $\mathrm{X} \Rightarrow \mathrm{Y}$ and gained no evidence for $\mathrm{X} \Leftarrow \mathrm{Y}$, because the reordered p-values are located in the uc-area. Moreover, in testing $\mathrm{X} \Rightarrow \mathrm{Y}$, nearly all p-values obtained from single tests on sub-time-series are close to 0, which showed that our single test on time series of size 100 in this example is already able to provide reliable results.

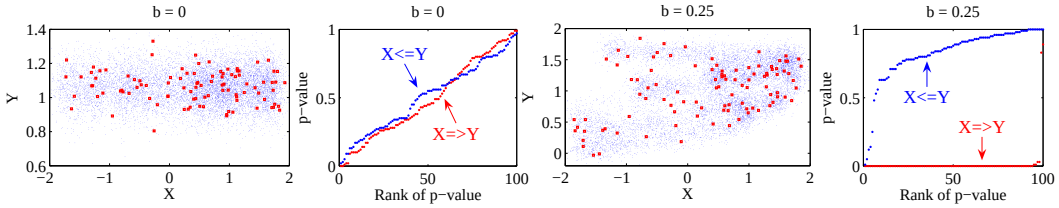


Figure 4: The first and third plot (from left) show bivariate time series of 10000 observations (fine points) generated by Hénon maps and random sub-time-series of 100 observations (bold points). The second and fourth plot show the corresponding Q-Q plots of p-values obtained by tests of predictability on 100 sub-time-series.

4.2 Logistic Maps

As the second simulated example, we consider the following pair of noisy logistic maps:

$$\begin{aligned} x_{t+1} &= (1-b_1)\, a\, x_t(1-x_t) + b_1\, a\, y_t(1-y_t) + \mu\, \xi_1 \\ y_{t+1} &= (1-b_2)\, a\, y_t(1-y_t) + b_2\, a\, x_t(1-x_t) + \mu\, \xi_2 \end{aligned}$$

represented as systems X and Y. $b_1, b_2 \in [0, 1]$ describe the coupling strengths between X and Y. If $b_1 = b_2 = 0$, X and Y are uncoupled; If $b_1, b_2 > 0$, we have a bi-directed coupling $\mathrm{X} \Leftrightarrow \mathrm{Y}$; If $b_1 = 0$ and $b_2 > 0$, we have a uni-directed coupling $\mathrm{X} \Rightarrow \mathrm{Y}$. The last case is also studied by Ancona et al. [7]. They claimed that in the noise-free case, i.e., $\mu = 0$, a transition to synchronization occurs at $b_2 = 0.37$ based on the calculation of the Lyapunov exponents. For this reason, we chose $b_1, b_2 < 0.37$. As proposed in [7], parameter a is fixed to 3.8; ξ_1, ξ_2 are unit variance Gaussian distributed noise terms; and μ is set at 0.01. We chose the start point (x_1, y_1) randomly from $[0, 1] \times [0, 1]$ and generated time series of length 10000 with $(b_1, b_2) \in \{(0, 0), (0, 0.3), (0.1, 0.3), (0.3, 0.3)\}$. We repeated the same experiment with 20 different start points that are randomly chosen in $[0, 1] \times [0, 1]$. All these time series were not observed to diverge. The resulting directions of Granger causality were always consistent.

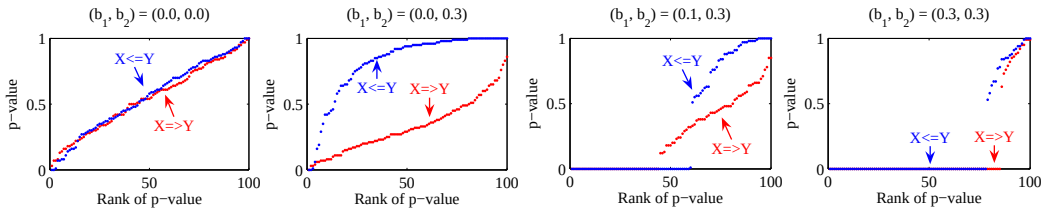


Figure 5: Q-Q plots of p-values obtained by tests of predictability on 100 random sub-time-series from time series generated by logistic maps.

Fig. 5 shows Q-Q plots of p-values based on 100 subsamples from one time series of length 10000 with various coupling strengths (b_1, b_2). In all 4 cases, our method identified the directions of coupling correctly: If $b_1 = b_2 = 0$, both $\mathrm{X} \Rightarrow \mathrm{Y}$ and $\mathrm{Y} \Rightarrow \mathrm{X}$ are rejected due to uniform distributions of p-values; If $(b_1, b_2) = (0, 0.3)$, $\mathrm{X} \Rightarrow \mathrm{Y}$ is accepted and $\mathrm{Y} \Rightarrow \mathrm{X}$ gained no evidence, which is

consistent with the underlying model. In the case of $(b_1, b_2) \in \{(0.1, 0.3), (0.3, 0.3)\}$, both $\mathrm{X} \Rightarrow \mathrm{Y}$ and $\mathrm{Y} \Rightarrow \mathrm{X}$ are accepted, which means the bi-directed Granger causality $\mathrm{Y} \Leftrightarrow \mathrm{X}$ is verified.

Interestingly, by means of two-sample-test, we can additionally confirm that when $b_1 = b_2 = 0.3$ the resulting p-values corresponding to testing $\mathrm{X} \Rightarrow \mathrm{Y}$ and to testing $\mathrm{Y} \Rightarrow \mathrm{X}$ are identically distributed, while in the case of $b_1 \neq b_2$ the resulting p-values corresponding to testing $\mathrm{X} \Rightarrow \mathrm{Y}$ and to testing $\mathrm{Y} \Rightarrow \mathrm{X}$ come from different distributions. This is reasonable, because the coupling is absolutely symmetric in X and Y, if $b_1 = b_2$. It seems plausible that the more right-skewed, the stronger the coupling (compare the case $b_1 < b_2$). But, we do not speculate on this property.

4.3 Co-movement of Stock Indexes

The analyzed raw dataset consists of daily closing values (adjusted for dividends and splits) of Dow Jones (DJ) industrial average index and NIKKEI 225 stock average index during the time between January 1984 and January 2008. Only days with trading activity in both stock exchanges were considered. The time increment in the raw data is not always exactly one day due to weekends and moving holidays. We transform the raw data into a series of day-to-day differences to describe the daily movements (DM) of indexes. After all, we have a bivariate time series with 5751 observations (Fig. 6, left).

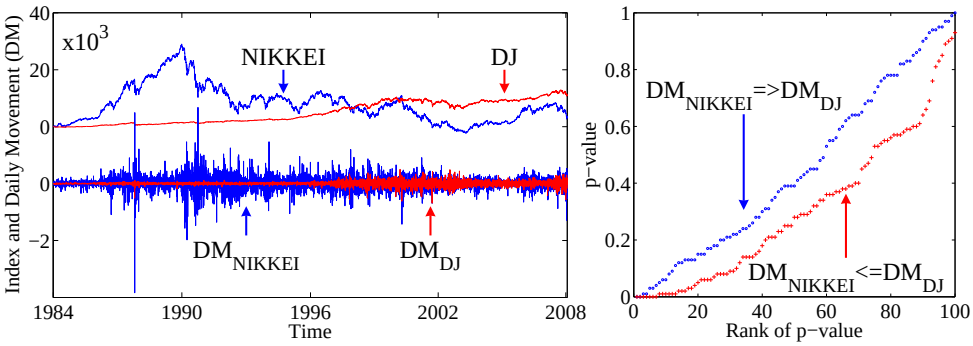


Figure 6: The left plot illustrates the time series of DJ and NIKKEI index and their daily movements (DM). The right plot is Q-Q plots of p-values obtained by testing predictability between "$\mathrm{DM}_{\mathrm{NIKKEI}}$" and "$\mathrm{DM}_{\mathrm{DJ}}$"on 100 random sub-time-series.

We ran our test procedure on this dataset. The p-values obtained for both causal hypotheses are visualized in the right plot of Fig. 6. The probability of the uniform distribution for the p-values corresponding to the hypothesis that the daily movement of DJ causes that of NIKKEI is $P^{\mathrm{Unif}} = 0.003 < \frac{1}{2}$ (p-values are positively skewed), and $P^{\mathrm{Unif}} = 0.875 \geq \frac{1}{2}$ for the reversed causal direction. Due to time difference, the US stock market actually opens after the market in Tokyo is already closed. For this reason, it is in fact justified to test the conditional predictability on Y_t by X_t ("X": $\mathrm{DM}_{\mathrm{NIKKEI}}$; "Y": $\mathrm{DM}_{\mathrm{DJ}}$) within one time slice. The resulting p-values obtained from tests of predictability on Y_t by X_t provides $P^{\mathrm{Unif}} = 1 \geq \frac{1}{2}$.

In summary, our testing procedure showed evidence of a uni-directed causality running from the daily movement of DJ to the daily movement of NIKKEI. The knowledge of the dynamics of DJ can significantly improve a prediction of the dynamics of NIKKEI, but the dynamics of NIKKEI has a very limited, yet non-significant impact on the future dynamics of DJ. The finding that the movement of DJ influences the movement of NIKKEI and not vice versa, which may seem trivial as a purely economical fact, but actually confirms in an independent way the validity of our kernel test formalism.

4.4 Cardiorespiratory Interaction

As another example of real-life systems, we consider the benchmark bivariate time series of heart rate and respiration force of a sleeping human suffering from sleep apnea (data set B of the Santa Fe Institute time series competition [23]) recorded in the sleep laboratory of the Beth Israel Hospital in

Boston, MA. The magnitudes considered are heart rate and respiration force. The data are plotted in Fig. 7 (left). The time interval between measurements is 0.5 seconds. As described in [24, 25], under normal, physiological conditions, the heart rate is modulated by respiration through a process known as Respiratory Sinus Arrhythmia (RSA). It is the natural cycle of arrhythmia that occurs through the influence of breathing on the flow of sympathetic and vagus impulses to the sinoatrial node of the heart. When we inhale, vagus nerve activity is impeded and the heart rate begins to increase. When we exhale, this pattern is reversed. This quasi-periodic modulation of heart rate by respiration is most notable in young, healthy subjects and decreases with age, which means "Heart $\Leftarrow$ Respiration".

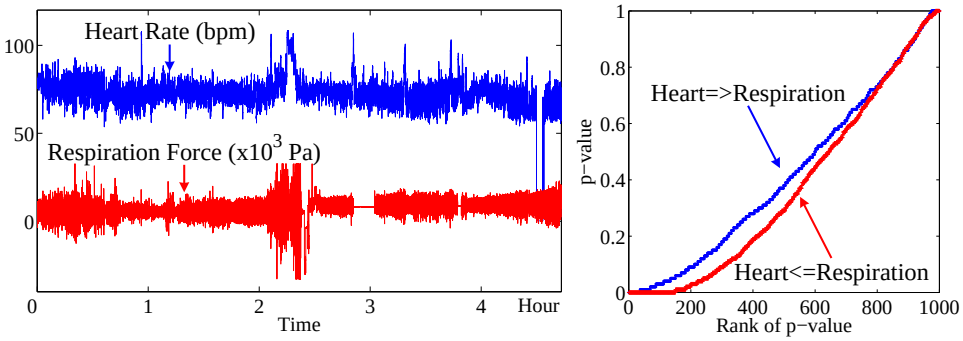


Figure 7: Time series of the heart rate and respiration force of a patient suffering sleep apnea (left). Q-Q plots of p-values obtained by testing predictability between "Heart" and "Respiration" on 1000 random sub-time-series (right).

However, this dataset corresponds to a patient suffering from sleep apnea, which is a breathing disorder characterized by brief interruptions of breathing during sleep. Sleep apnea affects the normal process of RSA, disturbing the usual patterns of interaction between the heart rate and respiration. As a result, the control of the heart rate by respiration becomes unclear. It may well be blocked, in accordance with the change in dynamics, that is characteristic of the so-called "dynamical diseases". Some studies [26, 22, 7] claimed a coupling in the reversed direction: "Heart $\Rightarrow$ Respiration". For these reasons, the bi-directed causation "Heart $\Leftrightarrow$ Respiration" might be likely the ground truth in this sample. The result of our test procedure is consistent with this prior knowledge, since for both directions we have $P^{\text{Unif}} = 0 < \frac{1}{2}$ (p-values are positively skewed). The identified bi-directed causation between heart rate and respiration suggests a probably causal link between sleep apnea and cardiovascular disease [27], although the exact mechanisms that underlie this relationship remain unresolved [28].

5 Conclusion

We have presented a kernel framework for detecting nonlinear Granger causality $\mathrm{X} \Rightarrow \mathrm{Y}$. We use the property that when the underlying process of X and Y are indeed uncoupled to each other, the p-values of testing prediction improvement of Y_{t+1} by X_t, given the relevant past observations $(\mathrm{Y}_t, \ldots, \mathrm{Y}_{t-l+1})$, are uniformly distributed over $[0, 1]$. The predictability improvement is captured by the trace norm of conditional covariance operators. In comparison to other nonlinear extensions of Granger causality as described in [6, 7, 29], our approach is designed in a more general kernel framework and can, in principle, be straightforwardly extended to analyzing multivariate time series.

References

[1] N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw. Geometry from a time series. *Physical Review Letters*, 45:712–716, 1980.

[2] M. Takens. Detecting strange attractors in turbulence. In D. A. Rand and L. S. Young, editors, *Dynamical Systems and Turbulence*, Lecture Notes in Mathematics, pages 366–381. Springer-Verlag, Berlin, 1982.

[3] M. B. Kennel, R. Brown, and H. D. I. Abarbanel. Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Physical Review A*, 45(6):3403–3411, 1992.

[4] A. M. Fraser and H. L. Swinney. Independent coordinates for strange attractors from mutual information. *Physical Review A*, 33(2):1134–1140, 1986.

[5] C. Granger. Investigating causal relations by econometric and cross-spectral methods. *Econometrica*, 37:424–438, 1969.

[6] Y. Chen, G. Rangarajan, J. Feng, and M. Ding. Analyzing multiple nonlinear time series with extended Granger causality. *Physics Letters A*, 324:26–35, 2004.

[7] N. Ancona, D. Marinazzo, and S. Stramaglia. Radial basis function approach to nonlinear granger causality of time series. *Physical Review E*, 70(5):066216.1–7, 2004.

[8] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404, 1950.

[9] C. Baker. Joint measures and cross-covariance operators. *Transactions of the American Mathematical Society*, 186:273–289, 1973.

[10] K. Fukumizu, F. Bach, and M. Jordan. Kernel dimension reduction in regression. Technical Report 715, Department of Statistics, University of California, Berkeley, CA, 2006.

[11] K. Fukumizu, A. Gretton, X. Sun, and B. Schölkopf. Kernel measures of conditional dependence. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Proceedings of the 21th Neural Information Processing Systems Conference*, Cambridge, MA, 2007. MIT Press. 489–496.

[12] B. Sriperumbudur, A. Gretton, K. Fukumizu, G. Lanckriet, and B. Schlkopf. Injective hilbert space embeddings of probability measures. In *Proceedings of the 21th Annual Conference on Learning Theory*, 2008. in press.

[13] I. Steinwart. On the influence of the kernel on the consistency of support vector machines. *Journal of Machine Learning Research*, 2:67–93, 2001.

[14] N. Ancona and S. Stramaglia. An invariance property of predictors in kernel-induced hypothesis spaces. *Neural Computation*, 18:749–759, 2006.

[15] M. Lukić and J. Beder. Stochastic processes with sample paths in reproducing kernel Hilbert spaces. *Transactions of the American Mathematical Society*, 353(10):3945–3969, 2001.

[16] C. Groetsch. *The theory of Tikhonov regularization for Fredholm equations of the first kind*. Pitman Publishing Program, Boston, MA, 1984.

[17] A. Hoerl and R. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[18] S. Fine and K. Scheinberg. Efficient SVM training using low-rank kernel representations. *Journal of Machine Learning Research*, 2:243–264, 2001.

[19] T. Schweder. Plots of p-values to evaluate many tests simultaneously. *Biometrika*, 69(3):493–502, 1982.

[20] Y. Hochberg. More powerful procedures for multiple significance testing. *Statistics in Medicine*, 9(7):811–818, 1990.

[21] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, et al. A kernel method for the two-sample-problem. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Proceedings of the 20th Neural Information Processing Systems Conference*, pages 513–520, Cambridge, MA, 2006. MIT Press.

[22] J. Bhattacharya, E. Pereda, and H. Petsche. Effective detection of coupling in short and noisy bivariate data. *IEEE Transactions on Systems, Man, and Cybernetics*, 33(1):85–95, 2003.

[23] D. Rigney, A. Goldberger, W. Ocasio, Y. Ichimaru, et al. Multi-channel physiological data: description and analysis (data set B). In A. Weigend and N. Gershenfeld, editors, *Time series prediction: Forecasting the future and understanding the past*, pages 105–129. Addison-Wesley, Reading, MA, 1993.

[24] Y. Ichimaru, K. Clark, J. Ringler, and W. Weiss. Effect of sleep stage on the relationship between respiration and heart rate variability. In *Proceedings of Computers in Cardiology 1990*, pages 657–660, Chicago, IL, 1990. IEEE Computer Society Press.

[25] P. Verdes. Assessing causality from multivariate time series. *Physical Review E*, 72(2):066222.1–9, 2005.

[26] T. Schreiber. Measuring information transfer. *Physical Review Letters*, 85(2):461–464, 2000.

[27] F. Roux, C. D'Ambrosio, and V. Mohsenin. Sleep-related breathing disorders and cardiovascular disease. *The American Journal of Medicine*, 108(5):396–402, 2000.

[28] H. Duchna, L. Grote, S. Andreas, R. Schulz, et al. Sleep-disordered breathing and cardio- and cerebrovascular diseases: 2003 update of clinical significance and future perspectives. *Somnologie*, 7(3):101–121, 2003.

[29] D. Marinazzo, M. Pellicoro, and S. Stramaglia. Nonlinear parametric model for Granger causality of time series. *Physical Review E*, 73(6):066216.1–6, 2006.

High Frequency Variability and Microstructure Bias

Adam Sykulski [*]
Dept. of Mathematics,
Imperial College London
SW7 2AZ, London
ams03@ic.ac.uk

Sofia Olhede [†]
Dept. of Statistical Science
University College London
WC1 E6BT, London
s.olhede@ucl.ac.uk

Grigorios Pavliotis [‡]
Dept. of Mathematics,
Imperial College London
SW7 2AZ, London
g.pavliotis@ic.ac.uk

Abstract

This paper treats the multiscale estimation of the integrated volatility of an Itô process immersed in high-frequency correlated noise. The multiscale structure of the problem is modelled explicitly, and the multiscale ratio is used to quantify energy contributions from the noise, estimated using the Whittle likelihood. This problem becomes more complex as we allow the noise structure greater flexibility, and multiscale properties of the estimation are discussed via a simulation study.

1 Introduction

The estimation of properties of continuous time stochastic processes, whose observation is immersed in high frequency nuisance structure is required in many different fields of application, for example molecular biology and finance. Various methods have been proposed to alleviate bias introduced into the estimation from high frequency nuisance structure, see for example [1–4]. Commonly the model of the observed process is as the process of interest X_{t_i} superimposed with noise ϵ_{t_i}, or

$$Y_{t_i} = X_{t_i} + \epsilon_{t_i}, \tag{1}$$

where Y_{t_i} is the observed process, X_{t_i} the unobserved component of interests, and ϵ_{t_i} is the microstructure noise effect. We model X_t, the process of interest with a suitable stochastic differential equation. For example, the Heston model is specified [5] by

$$dX_t = (\mu - \nu_t/2)\, dt + \sigma_t dB_t, \quad d\nu_t = \kappa\,(\alpha - \nu_t)\, dt + \gamma \nu_t^{1/2} dW_t, \tag{2}$$

where $\nu_t = \sigma_t^2$, and B_t and W_t are correlated 1-D Brownian motions.

Our main objective is to estimate the *integrated volatility*, $\langle X, X\rangle_T$ of the Itô process $\{X_t\}$, from the set of observations $\{Y_{t_i}\}$. Different methods have been proposed for determining the properties of X_{t_i}. An outstanding problem is proposing more robust inference methods. [3] has relaxed the assumptions of [1], to include inference of processes with jumps. Another possible direction of development is to include more complicated noise scenarios, namely allowing for correlation between observations. The main issue with such relaxations, is that as the permitted structure of X_t and ϵ_t become less stylized, it naturally becomes harder to separate energy due to the high frequency nuisance component from the process of interest.

Sykulski *et al.* [4] have proposed inference for multiscale processes based on using the discrete Fourier transform. Fourier domain estimators have also been used for estimating noisy Itô processes, see [6], but the main innovation of Sykulski *et al.* was to present a theoretical framework for Harmonizable processes [7, 8] of interest, and an automatic procedure for estimating the nuisance structure was proposed. The Whittle likelihood was used to estimate the energy level of the process of interest, as well as the noise contamination. The method was shown to perform well under various signal to noise scenarios, as well as path lengths, see [4].

[*] www.ecs.soton.ac.uk/people/as07r
[†] www.homepages.ucl.ac.uk/~ucaksc0/
[‡] www.ma.ic.ac.uk/~pavl/

The results of [4] or [1] are only appropriate when the noise is white. We shall in contrast in this paper discuss possible extensions of the multiscale estimators to the case of more complicated market microstructure, and illustrate the performance of the estimator in various noise scenarios.

2 Multiscale Estimation

In the absence of noise a suitable estimator of the integrated volatility, $\langle X,X\rangle_T = \int_0^T \sigma_t^2\,dt$, can be specified from the quadratic variation of the process $\{Y_t\}$. In the presence of market microstructure noise this is no longer true and it is necessary to employ a different estimation procedure. For ease of exposition we denote the difference process $Z_{t_i} - Z_{t_{i-1}}$ by $U_{t_i}^{(Z)}$ where $Z = X$, Y or ϵ. The Loève spectrum [7,8] of $U_{t_i}^{(Z)}$ will be denoted $S^{(Z)}(f_k, f_k)$, and we note that the observed quadratic variation can be rewritten as:

$$\widehat{\langle X,X\rangle}_T^{(b)} = \sum_{i=0}^{N-1}\left(U_{t_i}^{(Y)}\right)^2 = \sum_{k=-N/2}^{N/2-1}\left|J^{(Y)}(f_k)\right|^2 \tag{3a}$$

$$\widehat{S}^{(Y)}(f_k,f_k) = \left|J^{(Y)}(f_k)\right|^2, \quad J^{(Y)}(f_k) = \frac{1}{\sqrt{N}}\sum_{j=1}^{N-1} U_{t_j}^{(Y)} e^{-2\pi i t_j f_k}. \tag{3b}$$

with $f_k = \frac{k}{T}$. $\widehat{S}^{(Y)}(f_k,f_k)$ is the periodogram estimator, see [9], and normally has a single argument because the covariance of the Fourier Transform at two fixed frequencies is asymptotically equivalent to zero for a stationary process. We note directly from [4] that the bias of the estimator $\widehat{\langle X,X\rangle}_T^{(b)}$ is conveniently expressed in the Fourier domain by the observation that

$$\mathrm{E}\left\{\widehat{\langle X,X\rangle}_T^{(b)}\right\} = \sum_{k=-N/2}^{N/2-1} S^{(X)}(f_k,f_k) + \sigma_\epsilon^2 \sum_{k=-N/2}^{N/2-1} \left|2\sin(\pi f_k \Delta t)\right|^2 + O(N^\alpha) + O\left(N^{1-\alpha}\right). \tag{4}$$

The error terms follow from assumptions regarding the spectral properties of the process X_t, and are detailed in [4]. These assumptions determine the value of α. It is clear from eqn (4) that the influence of the noise increases for larger frequencies, and that the relative magnitude of $S^{(X)}(f_k,f_k)$ to $\sigma_\epsilon^2\left|2\sin(\pi f_k\Delta t)\right|^2$ at frequency f_k will determine the need for bias correction at f_k.

Sykulski *et al* proposed to measure the average energy of $U_t^{(X)}$ across frequencies, and determine the energy of $U_t^{(\epsilon)}$, using the form of the white noise spectrum. Despite $U_t^{(X)}$ assumed harmonizable and not necessarily stationary, with appropriate assumptions regarding the spectral correlation of the process, it is appropriate to use the Whittle likelihood, see [10], to determine the relative energy of the two processes across scales. Instead of using eqn (3b) to estimate the spectral contributions of the process of interest, a shrinkage estimator of $S^{(X)}(f_k,f_k)$ was therefore proposed in [4]:

$$\widehat{S}^{(X)}(f_k,f_k;L_k) = L_k\widehat{S}^{(Y)}(f_k,f_k). \tag{5}$$

$0 \le L_k \le 1$ is referred to as the 'multiscale ratio' and its optimal form for perfect bias correction when ϵ_{t_i} is white noise is given by:

$$L_k = \frac{S^{(X)}(f_k,f_k)}{S^{(X)}(f_k,f_k) + \sigma_\epsilon^2\left|2\sin(\pi f_k\Delta t)\right|^2}. \tag{6}$$

Of course this assumes perfect knowledge of $S^{(X)}(f_k,f_k)$ and is not a realizable estimator. Instead typical contributions of $S^{(X)}(f_k,f_k)$ across frequencies were considered, and the multiscale ratio replaced by a sort of average ratio corresponding to

$$\overline{L}_k = \frac{\sigma_X^2}{\sigma_X^2 + \sigma_\epsilon^2\left|2\sin(\pi f_k\Delta t)\right|^2}. \tag{7}$$

The justification for this choice is discussed in Sykulski *et al*. We estimate the parameters of $\overline{L}_k$ by maximising a pseudo-likelihood namely the multiscale Whittle likelihood defined in parameter

$\boldsymbol{\sigma} = \begin{pmatrix} \sigma_\epsilon^2 & \sigma_X^2 \end{pmatrix}$

$$\ell(\boldsymbol{\sigma}) = -\sum_{k=1}^{N/2-1} \log\left(\sigma_X^2 + \sigma_\epsilon^2 \left|2\sin(\pi f_k \Delta t)\right|^2\right) - \sum_{k=1}^{N/2-1} \frac{\widehat{S}^{(Y)}(f_k, f_k)}{\sigma_X^2 + \sigma_\epsilon^2 \left|2\sin(\pi f_k \Delta t)\right|^2}.$$

If $\left\{U_t^{(X)}\right\}$ is a stationary process, then the full Whittle likelihood (with σ_X^2 replaced by $S^{(X)}(f_k, f_k)$) will approximate the time-domain likelihood of the sample, under suitable regularity conditions, see [11].

The bias corrected estimator of the integrated volatility for an estimated $\hat{L}_K$ sequence becomes

$$\widehat{\langle X, X\rangle}_T^{(m_1)} = \sum_{k=-N/2}^{N/2-1} \widehat{S}^{(X)}(f_k, f_k; \widehat{L}_k). \tag{8}$$

In Sykulski *et al.* it was shown that the estimates of σ_X^2 and σ_ϵ^2 produced suitable $\widehat{L}_k$ such that bias corrected estimators of $S^{(X)}(f_k, f_k)$ with suitable properties were defined. Unfortunately it is not always reasonable to model the high frequency structure as white, and so more subtle modelling needs to be used when the noise is more complicated.

3 Correlated Noise

A key issue is treating correlation in the error terms. A reasonable relaxation of modelling ϵ_{t_i} as white would correspond to ϵ_{t_i} stationary. Stationary processes can be conveniently represented in terms of aggregations of uncorrelated white processes, using the Wold decomposition theorem [12][p. 187]. We may therefore write the zero-mean observation ϵ_{t_i} as

$$\epsilon_{t_i} = \sum_{j=0}^{\infty} \theta_{t_j} \eta_{t_i - t_j}, \tag{9}$$

where $\theta_{t_0} \equiv 1$, $\sum_j \theta_{t_j}^2 < \infty$, and $\{\eta_{t_n}\}$ satisfies $\mathrm{E}\left[\eta_{t_n}\right] = 0$ and $\mathrm{E}\left[\eta_{t_n}\eta_{t_m}\right] = \sigma_\eta^2 \delta_{n,m}$. Common practise would involve approximating the distribution by a finite number of elements in the sum, and thus truncate eqn (9) to $q \in \mathbb{Z}$. We therefore model the noise as a Moving Average (MA) process specified by

$$\epsilon_{t_i} = \eta_{t_i} + \sum_{k=1}^{q} \theta_{t_k} \eta_{t_{i-k}}, \tag{10}$$

and the spectral density function [9] of ϵ_{t_i} takes the form:

$$S^*(f; \boldsymbol{\theta}, \sigma_\eta^2) = \sigma_\eta^2 \left| 1 + \sum_{k=1}^{q} \theta_k e^{2i\pi f k} \right|^2. \tag{11}$$

In this case our spectral model for ϵ_{t_i} changes to a Loève spectrum of

$$S^{(\epsilon)}(f, f) = S^*(f; \boldsymbol{\theta}, \sigma_\eta^2) |2\sin(\pi f \Delta t)|^2. \tag{12}$$

Two possible methods now exist for treating the nuisance function of $S^*(f; \boldsymbol{\theta}, \sigma_\eta^2)$: we can use the method of Sykulski *et al.* directly without adjustment, assuming the variability of $S^*(f; \boldsymbol{\theta}, \sigma_\eta^2)$ to be moderate or we could adjust the methodology to encompass a parametric model for the noise, replacing $\sigma_\epsilon^2 |2\sin(\pi f \Delta t)|^2$ by $S^*(f; \boldsymbol{\theta}, \sigma_\eta^2) |2\sin(\pi f \Delta t)|^2$ when treating the frequency structure of the micro structure noise.

For a fixed and specified value of q, we may therefore estimate the parameters of the MA, using the Whittle likelihood, but where now $\sigma_\epsilon^2 |2\sin(\pi f \Delta t)|^2$ is replaced by $S^*(f) |2\sin(\pi f \Delta t)|^2$. We

thus get a multiscale likelihood [1] given with $\boldsymbol{\sigma} = \begin{pmatrix}\sigma_\eta^2 & \sigma_X^2\end{pmatrix}$ by

$$\ell(\boldsymbol{\sigma}, \boldsymbol{\theta}) = -\sum_{k=1}^{N/2-1} \log\left(\sigma_X^2 + S^*(f_k;\boldsymbol{\theta},\sigma_\eta^2)\left|2\sin(\pi f_k \Delta t)\right|^2\right) - \sum_{k=1}^{N/2-1} \frac{\widehat{S}^{(Y)}(f_k, f_k)}{\sigma_X^2 + S^*(f_k;\boldsymbol{\theta},\sigma_\eta^2)\left|2\sin(\pi f_k \Delta t)\right|^2}. \quad (13)$$

and the augmented multiscale ratio is defined by

$$\overline{L}_k^{(a)} = \frac{\sigma_X^2}{\sigma_X^2 + S^*(f_k;\boldsymbol{\theta},\sigma_\eta^2)\left|2\sin(\pi f_k \Delta t)\right|^2}. \quad (14)$$

If q is not assumed known, then model choice methods can also be applied to determine the value of q, such as applying the modified Akaike AIC [12][p. 287], and adding $2n(q+2)/[n-q-3]$ to minus two times the log multiscale likelihood, and minimizing this objective function. Some care must be applied as the Akaike AIC is known to overestimate the number of parameters, and BIC or some other model choice method may be applied. For a chosen value of q once we augment the estimation of σ_X^2 and σ_ϵ^2 with that of $\{\theta_{t_k}\}$, then we can estimate the noise spectrum and hence the multiscale ratio. This will yield an augmented estimator of the integrated volatility, replacing the parameters by their estimators in eqn (14), that we denote by $\widehat{L}_k^{(a)}$. Our new estimator then takes the form

$$\widehat{\langle X, X\rangle}_T^{(a)} = \sum_{k=-N/2}^{N/2-1} \widehat{S}^{(X)}(f_k, f_k; \widehat{L}_k^{(a)}). \quad (15)$$

This form both takes the high frequency structure into account, and permits the high frequency structure to be more dynamic than is the case of simple white noise nuisance structure.

4 Examples

We investigate the simple case of

$$\epsilon_{t_i} = \eta_{t_i} + \theta_1 \eta_{t_{i-1}}, \quad (16)$$

where then $S^*(f) = 1 + \theta_1^2 + 2\theta_1 \cos(2\pi f)$. Clearly it is of interest to investigate the effect of the variability of $S^*(f)$ on the multiscale estimation procedure. We note that setting $\theta_1 = 0$ recovers the white noise structure investigated in [4] and [1]. It is therefore of interest to compare our estimators over a range of values for θ_1 to study the effect of additional variability in the spectrum of the nuisance structure in the estimation of the integrated volatility. This is not a full study of the complete effects of complicated high-frequency structure superimposed on the process of interest: this study is intended to demonstrate the adverse effects of a more dynamic nuisance structure, and the potential of correcting for such effects using the multiscale structure of the process of interest.

We demonstrate the performance of our multiscale estimators of integrated volatility using the Heston model defined in eqn (2), with the same parameter values as used in [4] and [1], except this time we generate the microstructure noise process by eqn (16). Our new estimator $\widehat{\langle X, X\rangle}_T^{(a)}$, requires estimation of the parameters $(\sigma_X^2, \sigma_\epsilon^2, \theta_1)$ and this is done separately for each path using the MATLAB function fmincon on eqn (13). Figures 1(a) and 1(b) show the approximated σ_X^2 and $S^*(f_k;\theta_1,\sigma_\eta^2)\left|2\sin(\pi f_k \Delta t)\right|^2$ (in white) plotted over the periodograms $\widehat{S}^{(X)}(f_k, f_k)$ and $\widehat{S}^{(\epsilon)}(f_k, f_k)$ for one simulated path, where $\theta_1 = 0.5$. The parameters $(\sigma_X^2, \sigma_\epsilon^2, \theta_1)$ seem to have been approximated well, as the approximated spectral densities follow the shape of their respective periodograms. Figure 1(c) shows the corresponding multiscale ratio $\widehat{L}_k^{(a)}$ (in white) plotted over an unrealizable estimate of L_k:

$$\widetilde{L}_k = \frac{\widehat{S}^{(X)}(f_k, f_k)}{\widehat{S}^{(X)}(f_k, f_k) + \widehat{S}^{(\epsilon)}(f_k, f_k)}. \quad (17)$$

[1] Note that $\ell(\boldsymbol{\sigma}, \boldsymbol{\theta})$ is not strictly speaking a likelihood, see the full discussion in Sykulski et al. [4], but can for all intents and purposes be treated as such in this context.

Table 1: Root Mean Square Error (RMSE) for the different estimators of the integrated volatility, over different values of θ_1. The RMSEs are averaged over 7,500 paths.

RMSE$\{\cdot\}$	$\widehat{\langle X,X\rangle}_T^{(b)}$	$\widehat{\langle X,X\rangle}_T^{(s_1)}$	$\widehat{\langle X,X\rangle}_T^{(m_1)}$	$\widehat{\langle X,X\rangle}_T^{(a)}$	$\widehat{\langle X,X\rangle}_T^{(u)}$
$\theta_1 = -1$	3.51×10^{-2}	4.82×10^{-4}	7.35×10^{-5}	1.52×10^{-5}	1.46×10^{-5}
$\theta_1 = -0.75$	2.71×10^{-2}	3.62×10^{-4}	7.14×10^{-5}	1.56×10^{-5}	1.44×10^{-5}
$\theta_1 = -0.5$	2.05×10^{-2}	2.42×10^{-4}	6.40×10^{-5}	1.57×10^{-5}	1.44×10^{-5}
$\theta_1 = -0.25$	1.54×10^{-2}	1.21×10^{-4}	4.58×10^{-5}	1.60×10^{-5}	1.44×10^{-5}
$\theta_1 = 0$	1.17×10^{-2}	1.67×10^{-5}	1.61×10^{-5}	1.62×10^{-5}	1.43×10^{-5}
$\theta_1 = 0.25$	9.51×10^{-3}	1.22×10^{-4}	1.18×10^{-4}	1.67×10^{-5}	1.45×10^{-5}
$\theta_1 = 0.5$	8.78×10^{-3}	2.41×10^{-4}	4.67×10^{-4}	1.70×10^{-5}	1.43×10^{-5}
$\theta_1 = 0.75$	9.51×10^{-3}	3.62×10^{-4}	2.13×10^{-3}	1.74×10^{-5}	1.44×10^{-5}
$\theta_1 = 1$	1.17×10^{-2}	4.82×10^{-4}	9.82×10^{-3}	1.67×10^{-5}	1.45×10^{-5}

Our multiscale ratio provides a good estimate to L_k and will remove the noise microstructure from the correct frequencies by shrinkage. Figure 1(d) shows $\widehat{L}_k^{(a)}\widehat{S}^{(Y)}(f_k, f_k)$; the energy has been shrunk at frequencies affected by the microstructure noise and the spectrum is a good approximation to $\widehat{S}^{(X)}(f_k, f_k)$, which in turn should lead to a good approximation of the integrated volatility, compare with Figure 1(a). Figures 2(a) and 2(b) show two more estimated multiscale ratios $\widehat{L}_k^{(a)}$ (in white), but this time with $\theta_1 = -0.5$ and $\theta_1 = 1$ respectively. The multiscale estimator appears to correctly detect the correlation of noise in the process, as well as the magnitude of the signal to noise ratio. Note that for $\theta_1 = -0.5$ we shrink the estimated Loève spectrum at an increasing rate for high frequencies, whilst for $\theta_1 = 1$ we shrink in a highly non-monotone fashion across frequencies.

We investigate the performance of our new estimator against the estimators developed in [4] and [1] using Monte Carlo simulations. A range of values for θ_1 are used to investigate the effect of correlated noise. For each value of θ_1 we generated 7,500 simulated paths. Table I displays the results of our simulation, where the errors are calculated using a Riemann sum approximation on the X_t process (see [4] for details). Along with the performance of our new estimator $\widehat{\langle X,X\rangle}_T^{(a)}$ (eqn (15)), we include the performance of the estimator from [4], $\widehat{\langle X,X\rangle}_T^{(m_1)}$ (eqn (8)) and the best un-biased estimator developed in [1], $\widehat{\langle X,X\rangle}_T^{(s_1)}$. Naturally we do not aim to compare our estimator for correlated noise structure with that of [1, 4], as these were not developed for correlated noise, but more include these to show the necessity of treating correlation in the microstructure. Furthermore, had our Whittle estimators been sufficiently poor, then the variability of the estimated multiscale ratio would have made our proposed procedure unsuitable. We also include for reference, the biased estimator in eqn (3a), $\widehat{\langle X,X\rangle}_T^{(b)}$ (the quadratic variation on Y_t) and the unobservable unbiased estimator

$$\widehat{\langle X,X\rangle}_T^{(u)} = \sum_{i=0}^{N-1} \left(U_{t_i}^{(X)}\right)^2 \tag{18}$$

the quadratic variation on X_t, which in some sense is the best estimator that can be achieved.

The table shows that the new estimator performs remarkably well under different values of θ_1. In fact the RMSE of the estimator is very close to that of the unobservable quadratic variation, the best measure in the absence of market microstructure. The loss of efficiency by the more flexible model when $\theta_1 = 0$ is marginal whilst when $\theta_1 = 1$ the RMSE has decreased by a factor of a 500 compared to [4], and by a factor of 30 compared to [1], whilst if $\theta_1 = -1$ the RMSE has decreased by a factor of a 5 compared to [4], and by a factor of 30 compared to [1]. The small and consistent RMSE is due to the successful bias removal of the augmented multiscale estimator, where the low mean square error of the estimators of $(\sigma_X^2, \sigma_\epsilon^2, \theta_1)$, ensures that the bias in the estimated Loève spectrum of the process of interest is removed efficiently. Figure 3 shows the distribution of the estimates of these parameters over the 7,500 simulated paths for $\theta_1 = 0.5$; the estimation procedure is unbiased and has reasonably low variance.

The estimators $\widehat{\langle X,X\rangle}_T^{(s_1)}$ and $\widehat{\langle X,X\rangle}_T^{(m_1)}$ are inconsistent when additional structure is permitted in the noise. We stress that these are estimators based on assumptions of white noise, and their strong performance in this instance ($\theta_1 = 0$) is apparent. As we move away from white noise, $\widehat{\langle X,X\rangle}_T^{(s_1)}$ and $\widehat{\langle X,X\rangle}_T^{(m_1)}$ overcompensate for the noise when θ_1 in near minus one and undercompensate when θ_1 is near one. This happens because as the value of θ changes, taking values between minus one and one the spectral properties of the noise process change quite markedly with the appropriate shrinkage factor changing form in a corresponding fashion. For negative values of θ_1 the multiscale ratio, and smaller positive values of θ_1 the augmented multiscale ratio is decreasing at higher frequencies, whilst when θ_1 approaches one the multiscale ratio is not monotone (see Figures 1(c), 2(a) and 2(b)). $\widehat{\langle X,X\rangle}_T^{(m_1)}$ seems to still perform well for negative θ_1 values (note how the spectral form of the noise process is still largely the same shape) but performs disastrously for positive θ_1 values, due to the larger energy at lower frequencies that the estimator fails to remove. $\widehat{\langle X,X\rangle}_T^{(s_1)}$ suffers equivalent loss of performance as θ_1 moves away from zero in each direction; for such a time-domain estimator to perform better in these instances, the optimal subsampling rate of the estimator would have to be re-calibrated to incorporate the correlated noise. Nevertheless, all the estimators perform better than the noise polluted and biased estimator of the quadratic variation on Y_t, $\widehat{\langle X,X\rangle}_T^{(b)}$.

5 Conclusions

This paper has proposed extending the multiscale estimation methods of Sykulski *et al* for integrated volatility to include the case of stationary high frequency nuisance structure. It was found that naively applying estimators designed for the case of uncorrelated noise did not perform well. By modelling the nuisance structure as a Moving Average process, better bias correction could be applied at each frequency, and this substantially improved our estimator of the integrated volatility. Despite greater flexibility, the performance of the estimator did not deteriorate in terms of mean square error, which could have been a possible outcome. Note that the multiscale methods did not include parametric modelling of $\{X_t\}$ only approximating its multiscale nature. Future avenues of investigation includes rigorous model choice procedures, and the application of Bayesian estimation methods to naturally incorporate the multiscale ratio by Hierarchical modelling. Multiscale modelling shows great promise for designing inference methods for continuous time processes, by the increase in precision and power from investigating properties directly scale-by-scale.

References

[1] L. Zhang, P. A. Mykland, and Y. Ait-Sahalia, "A tale of two time scales: Determining integrated volatility with noisy high-frequency data", *J. Am. Stat. Assoc.*, vol. 100, pp. 1394–1411, 2005.

[2] G. A. Pavliotis and A. M. Stuart, "Parameter estimation for multiscale diffusions", *J. Stat. Phys.*, vol. 127, pp. 741–781, 2007.

[3] J. Fan and Y. Wang, "Multi-scale jump and volatility analysis for high-frequency financial data", *J. of the American Statistical Association*, vol. 102, pp. 1349–1362, 2007.

[4] A. Sykulski, S. C. Olhede, and G. Pavliotis, "Multiscale inference for high-frequency data", Tech. Rep. 290, Department of Statistical Science, University College London, arxiv.org/abs/0803.0392, 2008.

[5] Heston, "A closed form solution for options with stochastic volatility with applications to bond and currency options", *Review of Financial Studies*, vol. 6, pp. 327–343, 1993.

[6] M. E. Mancino and S. Sanfelici, "Robustness of fourier estimators of integrated volatility in the presence of microstructure noise", Tech. Rep., University of Firenze, 2006.

[7] M. Loève, *Probability theory. I*, Springer-Verlag, New York, fourth edition, 1977, Graduate Texts in Mathematics, Vol. 45.

[8] M. Loève, *Probability theory. II*, Springer-Verlag, New York, fourth edition, 1978, Graduate Texts in Mathematics, Vol. 46.

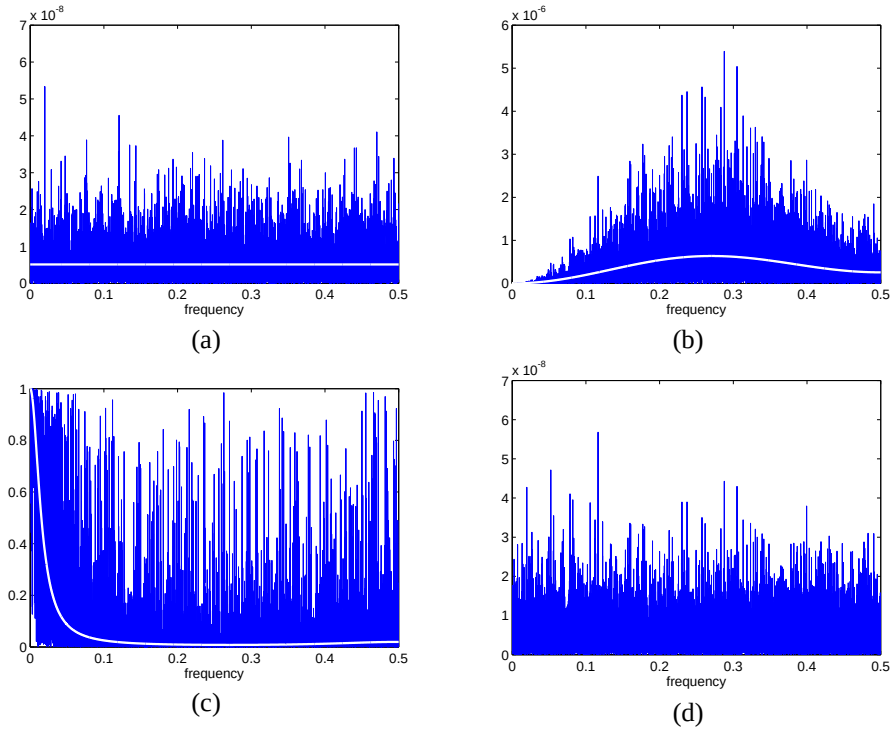


Figure 1: (a) The periodogram of a realisation of $U_t^{(X)}$ (solid line), (b) of a realisation of $U_t^{(\epsilon)}$ (solid line) with the Whittle estimates superimposed (white solid line), (c) the estimate of L_k from the raw periodograms of the unobserved processes (solid line) with the Whittle estimate $\widehat{L_k}$ superimposed (white solid line) and (d) the bias corrected estimator of the periodogram of $U_t^{(X)}$, using $\widehat{L}_k$. $\theta_1 = 0.5$ in this example. Notice the different scales in the four figures. Estimated spectra are here plotted on a linear scale for ease of comparison to the effect of applying L_k.

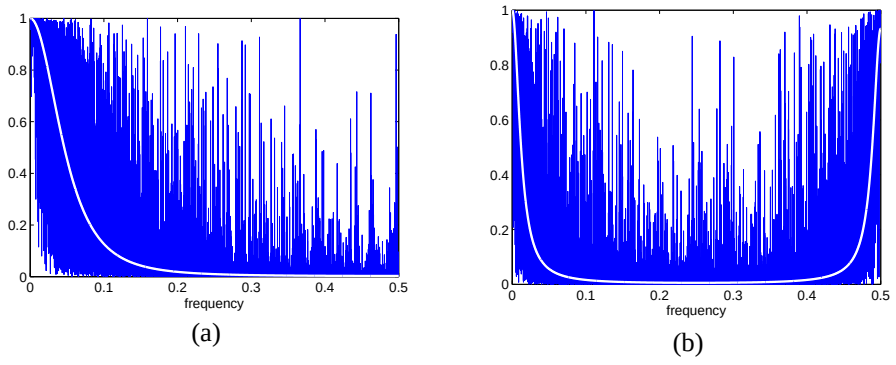
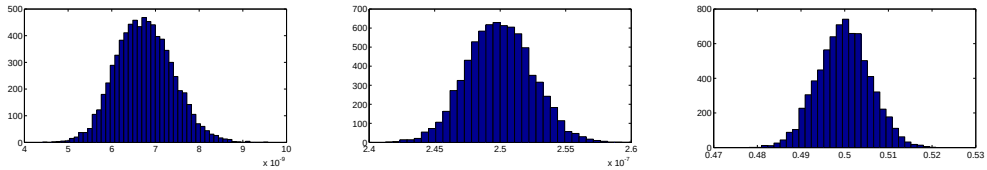


Figure 2: The estimate of L_k from the raw estimated spectra of the unobserved processes (solid line) with the Whittle estimate $\widehat{L_k}$ (white solid line) superimposed for (a) $\theta_1 = -0.5$ and (b) $\theta_1 = 1$. Notice the non-monotone structure of the multiscale ratio in the second case.



(a) (b) (c)

Figure 3: Histograms showing the distribution of the estimators of the parameters in $\widehat{L}_k$, for (a) σ_X^2, (b) σ_ϵ^2 and (c) θ_1 (where the true value of θ_1 is 0.5) over 7,500 simulated paths.

[9] D. Brillinger, *Time series: data analysis and theory*, Society for Industrial and Applied Mathematics, Philadelphia, USA, 2001.

[10] J. Beran, *Statistics for long-memory Processes*, Chapman and Hall, London, 1994.

[11] K. O Dzhamparidze and A. M. Yaglom, "Spectrum parameter estimation in time series analysis", in *Developments in Statistics*, PR Krishnaiah, Ed., vol. 4, pp. 1–181. New York: Academic Press., 1983.

[12] Peter J. Brockwell and Richard A. Davis, *Time Series: Theory and Methods*, Springer, Berlin, Germany, 1991.

Markov Chain Monte Carlo Algorithms for Gaussian Processes

Michalis K. Titsias*
School of Computer Science
University of Manchester
Manchester M13 9PL, UK
mtitsias@cs.man.ac.uk

Neil Lawrence
School of Computer Science
University of Manchester
Manchester M13 9PL, UK
Neil.Lawrence@manchester.ac.uk

Magnus Rattray
School of Computer Science
University of Manchester
Manchester M13 9PL, UK
magnus@cs.man.ac.uk

Abstract

We discuss Markov chain Monte Carlo algorithms for sampling functions in Gaussian process models. A first algorithm is a local sampler that iteratively samples each local part of the function by conditioning on the remaining part of the function. The partitioning of the domain of the function into regions is automatically carried out during the burn-in sampling phase. A more advanced algorithm uses control variables which are auxiliary function values that summarize the properties of the function. At each iteration, the algorithm proposes new values for the control variables and then generates the function from the conditional Gaussian process prior. The control input locations are found by minimizing the total variance of the conditional prior. We apply these algorithms to estimate non-linear differential equations in Systems Biology.

1 Introduction

Gaussian processes (GPs) are used for Bayesian non-parametric estimation of unobserved or latent functions. In regression problems with Gaussian likelihoods, GP models are analytically tractable, while for classification deterministic approximate algorithms are widely used [16, 3, 6, 10]. However, in recent applications of GP models in Systems Biology [1] that require the estimation of ordinary differential equations [2, 12, 7], the development of deterministic approximations is difficult, since the likelihood can be highly complex. Furthermore, accurate estimation in Systems Biology models is important and can facilitate a reliable Bayesian ranking of alternative models [15]. In this paper, we consider MCMC algorithms for doing inference in GP models. The advantage of MCMC over deterministic approximate inference is that it provides exact answers in the limit of long runs.

We introduce two sampling algorithms that construct the proposal distributions by utilizing the GP prior. The first algorithm is a local sampler that iteratively samples each local part of the function by conditioning on the remaining part of the function. Local sampling is implemented by iteratively generating samples from conditional GP prior distributions. The partitioning of the function points into groups is determined during the burn-in phase of MCMC using an hierarchical clustering process. The second algorithm is a global sampler that uses control variables. The control variables are auxiliary function values that summarize the properties of the function. At each iteration, the algorithm proposes new values for the control variables and samples the function by drawing from the conditional GP prior (given the proposed values for the control variables). The control input locations are found by continuously minimizing an objective function. This function is the least squares

*http://www.cs.man.ac.uk/∼mtitsias/.

error of reconstructing the latent function values from the control variables which also equals to the total variance of the conditional GP prior.

We apply the MCMC algorithms to infer biological networks where a set of genes are regulated by a transcription factor protein [7, 4]. The relationship between the protein and the target genes is governed by a non-linear system of ordinary differential equations where the concentration of the protein is an unobserved time continuous function. Given a set of gene expression mRNA measurements and assuming a GP prior over the protein concentration, we apply Bayesian inference using MCMC. We also compare the algorithms with Gibbs sampling in standard regression problems.

2 GP models

Assume a set of inputs $(\mathbf{x}_1, \ldots, \mathbf{x}_N)$ and a set of function values $\mathbf{f} = (f_1, \ldots, f_N)$ evaluated at those inputs. A Gaussian process places a prior on $\mathbf{f}$ which is a N-dimensional Gaussian distribution so as $p(\mathbf{f}) = N(\mathbf{y}|\boldsymbol{\mu}, K)$. The mean $\boldsymbol{\mu}$ is typically zero and the covariance matrix K is defined by the kernel function $k(\mathbf{x}_n, \mathbf{x}_m)$ that depends on parameters $\boldsymbol{\theta}$.

GPs are widely used for supervised learning [10]. Given a set of observed pairs $(\mathbf{y}_i, \mathbf{x}_i)$, where $i = 1, \ldots, N$, we assume a likelihood model $p(\mathbf{y}|\mathbf{f})$ that depends on parameters $\boldsymbol{\alpha}$ and associates the data with the latent function $\mathbf{f}$. For regression or classification problems, the latent function values are evaluated at the observed inputs and the likelihood factorizes according to $p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^{N} p(y_i|f_i)$. However, for other type of applications, such as modelling latent functions in ordinary differential equations, the above factorization is not applicable. Assuming that we have obtained suitable values for the model parameters $(\boldsymbol{\theta}, \boldsymbol{\alpha})$ inference over the latent function values $\mathbf{f}$ is done by applying Bayes rule:

$$p(\mathbf{f}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{f})p(\mathbf{f}). \tag{1}$$

For regression, where the likelihood is Gaussian, the above posterior is a Gaussian distribution that can be obtained using simple algebra. When the likelihood $p(\mathbf{y}|\mathbf{f})$ is non-Gaussian, computations become intractable and we need to consider approximations. Next we discuss MCMC algorithms that are applied independently from the functional form of the likelihood.

3 Sampling algorithms for GP models

The MCMC algorithm we consider is the general Metropolis-Hastings (MH) algorithm [11]. Suppose we wish to sample from the posterior in eq. (1). The MH algorithm forms a Markov chain. We initialize $\mathbf{f}^{(0)}$ and we consider a proposal distribution $Q(\mathbf{f}^{(t+1)}|\mathbf{f}^{(t)})$ that allows us to draw a new state given the current state. The new state is accepted with probability $\min(1, A)$ where

$$A = \frac{p(\mathbf{y}|\mathbf{f}^{(t+1)})p(\mathbf{f}^{(t+1)})}{p(\mathbf{y}|\mathbf{f}^{(t)})p(\mathbf{f}^{(t)})} \frac{Q(\mathbf{f}^{(t)}|\mathbf{f}^{(t+1)})}{Q(\mathbf{f}^{(t+1)}|\mathbf{f}^{(t)})}. \tag{2}$$

To apply this generic algorithm, we need to choose the proposal distribution Q. For GP models, finding a good proposal distribution is challenging since $\mathbf{f}$ is high dimensional and the posterior distribution can be highly correlated.

To motivate the algorithms presented in sections 3.1 and 3.2, we discuss two extreme options for specifying the proposal distribution Q. One simple way to choose Q is to set it equal to the GP prior $p(\mathbf{f})$. This gives us an independent MH algorithm [11]. However, sampling from the GP prior is very inefficient as it is unlikely to obtain a sample that will fit the data. Thus the Markov chain will get stuck in the same state for thousands of iterations. On the other hand, sampling from the prior is appealing because any generated sample satisfies the smoothness requirement imposed by the covariance function. Functions drawn from the posterior GP process should satisfy the same smoothness requirement as well.

The other extreme choice for the proposal , that has been considered in [8], is to apply Gibbs sampling where we iteratively draw samples from each posterior conditional density $p(f_i|\mathbf{f}_{-i}, \mathbf{y})$ with $\mathbf{f}_{-i} = \mathbf{f} \setminus f_i$. However, Gibbs sampling can be extremely slow for densely discretized functions, as in the regression problem of Figure 1, where the posterior GP process is highly correlated. To clarify this, note that the variance of the posterior conditional $p(f_i|\mathbf{f}_{-i}, \mathbf{y})$ is smaller or equal to the

variance of the conditional GP prior $p(f_i|\mathbf{f}_{-i})$. However, $p(f_i|\mathbf{f}_{-i})$ may already have a tiny variance caused by the conditioning on all remaining latent function values. The more densely discretized a function is (relative to the manifold of the input data), the more inefficient Gibbs sampling becomes since the variance of $p(f_i|\mathbf{f}_{-i})$ tends to zero. For the one-dimensional example in Figure 1, Gibbs sampling is practically not applicable. We further study this issue in section 5.

A similar algorithm to Gibbs sampling can be expressed by using as a proposal distribution the sequence of the conditional densities $p(f_i|\mathbf{f}_{-i})$[1]. We call this algorithm the Gibbs-like algorithm. This algorithm can exhibit a high acceptance rate, but it is inefficient to sample from highly correlated functions. Next we describe a modification of the Gibbs-like algorithm that is more efficient.

3.1 Sampling using local regions

To overcome the limitations of the Gibbs and Gibbs-like algorithm we can divide the domain of the function into regions and sample the entire function within each region. Assuming that the number of the regions depends mainly on the shape of the function and not on the discretization, this scheme can be more efficient.

Let $\mathbf{f}_k$ denote the function values that belong to the local region k, where $k = 1, \ldots, K$ and $\mathbf{f}_1 \cup \ldots \cup \mathbf{f}_K = \mathbf{f}$. New values for the region k are proposed by drawing from the conditional GP prior $p(\mathbf{f}_k^{t+1}|\mathbf{f}_{-k}^{(t)})$, where $\mathbf{f}_{-k} = \mathbf{f} \setminus \mathbf{f}_k$, by conditioning on the remaining function values. $\mathbf{f}_k^{(t+1)}$ is accepted with probability $\min(1, A)$ where

$$A = \frac{p(\mathbf{y}|\mathbf{f}_k^{(t+1)}, \mathbf{f}_{-k}^{(t)})}{p(\mathbf{y}|\mathbf{f}_k^{(t)}, \mathbf{f}_{-k}^{(t)})}. \tag{3}$$

Sampling $\mathbf{f}_k$ is iterated between all different regions $k = 1, \ldots, K$. Note that the terms associated with the GP prior cancel out from the acceptance probability since sampling from the conditional prior ensures that any proposed sample is consistent with the prior smoothness requirement. Sampling from the GP prior and the Gibbs-like algorithm are special cases of the above algorithm.

To apply the above algorithm, we need to partition the function values $\mathbf{f}$ into clusters. This process of adapting the proposal distribution can be carried out during the burn-in sampling phase. If we start with a small number of clusters, so as the acceptance rate is very low, our objective is to refine these initial clusters in order to increase the acceptance. Following the widely used heuristics [5] according to which desirable acceptance rates of MH algorithms are around $1/4$, we require the algorithm to sample with acceptance rate larger than $1/4$.

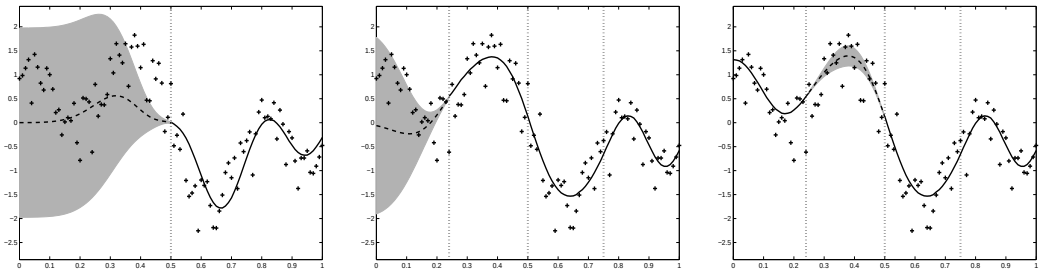
We obtain a initial partitioning of the vector $\mathbf{f}$ by clustering the inputs X using the kmeans algorithm. Then we start the simulation and we observe the local acceptance rate r_k associated with the proposal $p(\mathbf{f}_k|\mathbf{f}_{-k})$. Each r_k provides information about the variance of the proposal distribution relative to the local characteristics of the function. A small r_k implies that $p(\mathbf{f}_k|\mathbf{f}_{-k})$ has high variance and most of the generated samples are outside of the support of the GP posterior process; see Figure 1 for an illustrative example. Thus, when r_k is small, we split the cluster k into two clusters by locally applying the kmeans algorithm using all the inputs previously assigned to the initial cluster k. Clusters that have high acceptance rate are unchanged. This hierarchical partitioning process is recursively repeated until all the current clusters exhibit a local acceptance rate larger than a predefined threshold (this was set to $1/4$ for all our experiments). The above partitioning process is supervised since the information provided by the MH steps is used to decide which clusters need to be split into smaller clusters.

Once the adaption of the proposal distribution is ended, we can start sampling from the posterior GP process. The final form of the proposal distribution is a partition of the vector $\mathbf{f}$ into K disjoint groups and the conditional GP prior is the proposal distribution for each group.

3.2 Sampling using control points

The algorithm described previously is a local sampler that samples each part of the function by conditioning on the remaining part of the function. A limitation of this approach is that the variance

[1]Thus we replace the proposal distribution $p(f_i|\mathbf{f}_{-i}, \mathbf{y})$ with the prior conditional $p(f_i|\mathbf{f}_{-i})$.



(a) (b) (c)

Figure 1: Illustration of the hierarchical clustering process. The panel in (a) shows the variance (displayed with shaded two standard errors bars) of the initial conditional GP prior where we condition on the right side of the function. Since the variance is high the generated local parts of the function will not fit the data often. Dividing the local input region in (a) into two smaller groups (plots (b) and (c)) results a decrease of the variance of the newly formed GP conditional priors and an increase of the acceptance rate.

of the proposal distributions can be small close to the boundaries between neighbouring function regions; see Figure 1. This can result in a slow exploration of the probability density mass. In this section we discuss a different MH algorithm that can sample the whole function at once.

Let $\mathbf{f}_c$ be a set of K auxiliary function values that are evaluated at inputs X_c and drawn from the GP prior. $\mathbf{f}_c$ are called the control variables and their meaning is analogous to the inducing variables used in sparse GP models; see e.g. [13, 14, 9]. To compute the posterior $p(\mathbf{f}|\mathbf{y})$ based on control variables we use the expression

$$p(\mathbf{f}|\mathbf{y}) = \int_{\mathbf{f}_c} p(\mathbf{f}|\mathbf{f}_c, \mathbf{y}) p(\mathbf{f}_c|\mathbf{y}) d\mathbf{f}_c. \tag{4}$$

Assuming that $\mathbf{f}_c$ is highly informative about $\mathbf{f}$, so as $p(\mathbf{f}|\mathbf{f}_c, \mathbf{y}) \simeq p(\mathbf{f}|\mathbf{f}_c)$, we can approximately sample from $p(\mathbf{f}|\mathbf{y})$ in a two-stage manner: firstly sample the control variables from $p(\mathbf{f}_c|\mathbf{y})$ and then generate $\mathbf{f}$ from the conditional prior $p(\mathbf{f}|\mathbf{f}_c)$. This scheme can allow us to introduce a MH algorithm, where we need to specify only a proposal distribution $q(\mathbf{f}_c^{(t+1)}|\mathbf{f}_c^{(t)})$, that will mimic sampling from $p(\mathbf{f}_c|\mathbf{y})$, and always sample $\mathbf{f}$ from the conditional prior $p(\mathbf{f}|\mathbf{f}_c)$. The whole proposal distribution takes the form

$$Q(\mathbf{f}^{(t+1)}, \mathbf{f}_c^{(t+1)}|\mathbf{f}^{(t)}, \mathbf{f}_c^{(t)}) = p(\mathbf{f}^{(t+1)}|\mathbf{f}_c^{(t+1)}) q(\mathbf{f}_c^{(t+1)}|\mathbf{f}_c^{(t)}). \tag{5}$$

Figure 2 illustrates the steps of sampling from this distribution. Each proposed sample is accepted with probability $\min(1, A)$ where A is given by

$$A = \frac{p(\mathbf{y}|\mathbf{f}^{(t+1)}) p(\mathbf{f}_c^{(t+1)})}{p(\mathbf{y}|\mathbf{f}^{(t)}) p(\mathbf{f}_c^{(t)})} . \frac{q(\mathbf{f}_c^{(t)}|\mathbf{f}_c^{(t+1)})}{q(\mathbf{f}_c^{(t+1)}|\mathbf{f}_c^{(t)})}. \tag{6}$$

The usefulness of the above sampling scheme stems from the fact that the control variables can form a low-dimensional representation of the function. Assuming that these variables are much fewer than the points in $\mathbf{f}$, the sampling is mainly carried out in the low dimensional space. In section 3.3 we describe how to select the number K of control variables and the inputs X_c so as $\mathbf{f}_c$ becomes highly informative about $\mathbf{f}$. In the remainder of this section we discuss how we set the proposal distribution $q(\mathbf{f}_c^{(t+1)}|\mathbf{f}_c^{(t)})$.

A suitable choice for q is to use a Gaussian distribution with diagonal or full covariance matrix. The covariance matrix can be adapted during the burn-in phase of MCMC in order to increase the acceptance rate. Although this scheme is general, it has practical limitations. Firstly, tuning a full covariance matrix is time consuming and in our case this adaption process must be carried out simultaneously with searching for an appropriate set of control variables. Also, since the terms involving $p(\mathbf{f}_c)$ do not cancel out in the acceptance probability in eq. (6), using a diagonal covariance for the q distribution has the risk of proposing control variables that may not satisfy the GP prior smoothness requirement. To improve on these issues, we define q by utilizing the GP

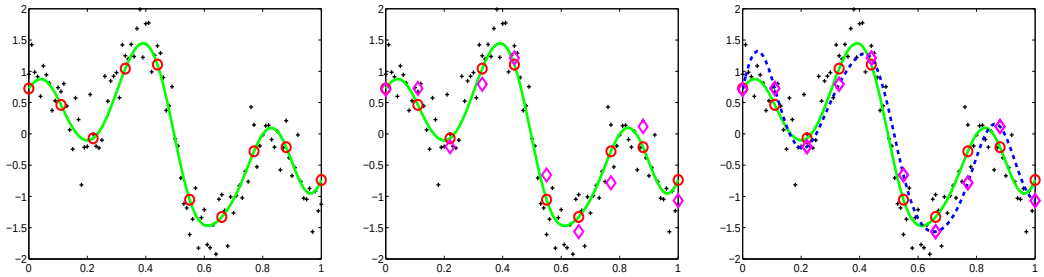


Figure 2: Illustration of sampling using control variables. (left) shows the current GP function $\mathbf{f}^{(t)}$ with green, the data and the current location of the control points (red circles). (middle) shows the proposed new positions for the control points (circles in magenda). (right) shows the new function values $\mathbf{f}^{(t+1)}$ drawn from the conditional GP prior (blue dotted line).

prior. According to eq. (4) a suitable choice for q must mimic the sampling from the posterior $p(\mathbf{f}_c|\mathbf{y})$. Given that the control points are far apart from each other, Gibbs sampling in the control variables space can be efficient. However, iteratively sampling f_{c_i} from the conditional posterior $p(\mathbf{f}_{c_i}|\mathbf{f}_{c_{-i}}, \mathbf{y}) \propto p(\mathbf{y}|\mathbf{f}_c)p(f_{c_i}|\mathbf{f}_{c_{-i}})$, where $\mathbf{f}_{c_{-i}} = \mathbf{f}_c \setminus f_{c_i}$ is intractable for non-Gaussian likelihoods[2]. An attractive alternative is to use a Gibbs-like algorithm where each f_{c_i} is drawn from the conditional GP prior $p(\mathbf{f}_{c_i}^{(t+1)}|\mathbf{f}_{c-i}^{(t)})$ and is accepted using the MH step. More specifically, the proposal distribution draws a new $f_{c_i}^{(t+1)}$ for a certain control variable i from $p(\mathbf{f}_{c_i}^{(t+1)}|\mathbf{f}_{c-i}^{(t)})$ and generates the function $\mathbf{f}^{(t+1)}$ from $p(\mathbf{f}^{(t+1)}|\mathbf{f}_{c_i}^{(t+1)}, \mathbf{f}_{c_{-i}}^{(t)})$. The sample $(f_{c_i}^{(t+1)}, \mathbf{f}^{(t+1)})$ is accepted using the MH step. This scheme of sampling the control variables one-at-a-time and resampling $\mathbf{f}$ is iterated between different control variables. A complete iteration of the algorithm consists of a full scan over all control variables. The acceptance probability A in eq. (6) becomes the likelihood ratio and the prior smoothness requirement is always satisfied.

Although, the control variables are sampled one-at-at-time, $\mathbf{f}$ can still be drawn with a considerable variance in all regions in the input space apart from the regions close to the control variables that are kept fixed. A full scan over all control variables can allow the function to significantly change everywhere.

3.3 Selection of the control variables

To apply the previous algorithm we need to select the number K of the control points and the associated inputs X_c. X_c must be chosen so that knowledge of $\mathbf{f}_c$ can determine $\mathbf{f}$ with small error. The prediction of $\mathbf{f}$ given $\mathbf{f}_c$ is equal to $K_{f,c}K_{c,c}^{-1}\mathbf{f}_c$ which is the mean of the conditional prior $p(\mathbf{f}|\mathbf{f}_c)$. A suitable way to search over X_c is to minimize the reconstruction error $||\mathbf{f} - K_{f,c}K_{c,c}^{-1}\mathbf{f}_c||^2$ averaged over any possible value of $(\mathbf{f}, \mathbf{f}_c)$:

$$G(X_c) = \int_{\mathbf{f},\mathbf{f}_c} ||\mathbf{f} - K_{f,c}K_{c,c}^{-1}\mathbf{f}_c||^2 p(\mathbf{f}|\mathbf{f}_c)p(\mathbf{f}_c)d\mathbf{f}d\mathbf{f}_c = \mathrm{Tr}(K_{f,f} - K_{f,c}K_{c,c}^{-1}K_{f,c}^T).$$

The quantity inside the trace is the covariance matrix of $p(\mathbf{f}|\mathbf{f}_c)$ and thus $G(X_c)$ is the total variance of this distribution. We can minimize $G(X_c)$ w.r.t. X_c using continuous optimization. Note that $G(X_c)$ is nonnegative and when it becomes zero, $p(\mathbf{f}|\mathbf{f}_c)$ becomes a delta function.

To find the number K of control points we minimize $G(X_c)$ by incrementally adding control variables until the total variance of $p(\mathbf{f}|\mathbf{f}_c)$ becomes smaller than a certain percentage of the total variance of the prior $p(\mathbf{f})$. 5% was the threshold used in all our experiments. Then we start the simulation and we observe the acceptance rate of the Markov chain. According to the heuristics [5] which suggest that desirable acceptance rates of MH algorithms are around $1/4$, we require a full iteration of the algorithm (a complete scan over the control variables) to have an acceptance rate larger than $1/4$[3]. When for the current set of control inputs X_c the chain has a low acceptance rate,

[2]This is because we need to integrate out $\mathbf{f}$ in order to compute $p(\mathbf{y}|\mathbf{f}_c)$.

[3]This means that the acceptance rate of each control variable i must be larger than $\frac{1}{4K}$.

it means that the variance of $p(\mathbf{f}|\mathbf{f}_c)$ is still too high and we need to add more control points in order to further reduce $G(X_c)$. The process of observing the acceptance rate and adding control variables is continued until we reach the desirable acceptance rate.

When the training inputs X are placed uniformly in the space, and the kernel function is stationary, the minimization of G places X_c in a regular grid, as happens in the example of Figure 2. In general, the minimization of G places the control inputs close to the clusters of the input data in such a way that the kernel function is taken into account. This also suggests that G can be used for learning inducing variables in sparse GP models in a unsupervised fashion where the observed outputs $\mathbf{y}$ are not involved.

4 Transcriptional regulation

In this section we consider a small biological sub-system where a set of target genes are regulated by one transcription factor (TF) protein. Ordinary differential equations (ODEs) can provide an useful framework for modelling the dynamics in these biological networks [1, 2, 12, 7, 4]. The concentration of the TF and the gene specific kinetic parameters are typically unknown and need to be estimated by making use of a set of observed gene expression levels. We use a GP prior to model the unobserved TF activity, as proposed in [7], and apply full Bayesian inference based on the MCMC algorithm presented previously.

Barenco et al. [2] introduce a linear ODE model for gene activation from TF. This approach was extended in [12, 7, 4] to account for non-linear models. The general form of the ODE model for transcription regulation with a single TF has the form

$$\frac{dy_j(t)}{dt} = B_j + S_j g(f(t)) - D_j y_j(t), \tag{7}$$

where the changing level of a gene j's expression, $y_j(t)$, is given by a combination of basal transcription rate, B_j, sensitivity, S_j, to its governing TF's activity, $f(t)$, and the decay rate of the mRNA, D_j. The differential equation can be solved for $y_j(t)$ giving

$$x_j(t) = \frac{B_j}{D_j} + A_j e^{-D_j t} + S_j e^{-D_j t} \int_0^t g(f(u)) e^{D_j u} du, \tag{8}$$

where A_j term arises from the initial condition. Due to the non-linearity of the g function that transforms the TF, the integral in the above expression is not analytically obtained. However, numerical integration can be efficiently used to estimate this quantity as follows. Assuming a very dense grid $(u_i)_{i=1}^P$ of points in the time axis and discretizing the TF according to $f_p = f(u_p)$, equation (8) is written as

$$y_j(t) = \frac{B_j}{D_j} + A_j e^{-D_j t} + S_j e^{-D_j t} \sum_{p=1}^{P_t} w_p g(f_p) e^{D_j u_p}, \tag{9}$$

where the weights w_p arise from the numerical integration method used and, for example, can be given by the composite Simpson rule.

The transcription protein $f(t)$ in the above system of ODEs is a latent function that needs to be estimated. Additionally the kinetic parameters of each gene $\boldsymbol{\alpha}_j = (B_j, D_j, S_j, A_j)$ are unknown and need to be estimated as well. To infer these quantities we use mRNA measurements (obtained from microarray experiments) of N target genes at T different time steps. Let y_{jt} denote the observed gene expression level of gene j at time t and let $\mathbf{y} = \{y_{jt}\}$ collect together all these observations. Assuming a Gaussian noise for the observed gene expressions the likelihood of our data has the form

$$p(\mathbf{y}|\mathbf{f}, \{\boldsymbol{\alpha}_j\}_{j=1}^N) = \prod_{j=1}^N \prod_{t=1}^T p(y_{jt}|\mathbf{f}_{1\leq p\leq P_t}, \boldsymbol{\alpha}_j), \tag{10}$$

where each probability density in the above product is a Gaussian with mean given by eq. (9) and $\mathbf{f}_{1\leq p\leq P_t}$ denotes the TF values up to time t. Notice that this likelihood is non-Gaussian due to the non-linearity of g. Further, this likelihood does not have a factorized form, as in the regression and classification cases, since an observed gene expression depends on the protein concentration activity

in all previous times points. Also note that the discretization of the TF in P time points corresponds to a very dense grid, while the gene expression measurements are sparse, i.e. $P \gg T$.

To apply full Bayesian inference in the above model, we need to define prior distributions over all unknown quantities. The protein concentration $\mathbf{f}$ is a positive quantity, thus a suitable prior is to consider a GP prior for $\log \mathbf{f}$. The kinetic parameters of each gene are all positive scalars. Those parameters are given vague gamma priors. Sampling the GP function is done exactly as described in section 3; we have only to plug in the likelihood from eq. (10) in the MH step. Sampling from the kinetic parameters is carried using Gaussian proposal distributions with diagonal covariance matrices that sample the positive kinetic parameters in the log space.

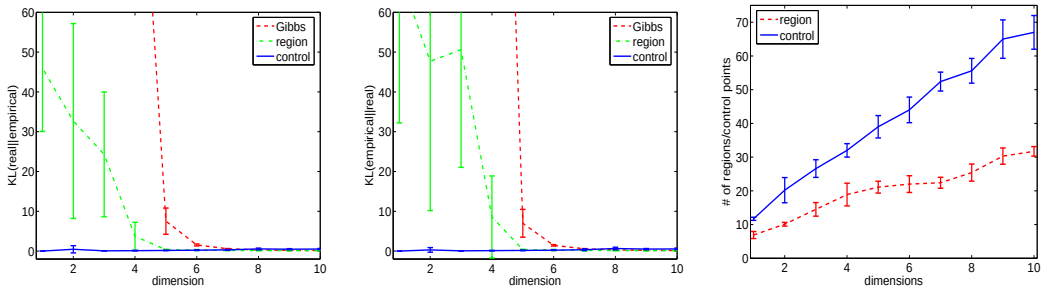
5 Experiments

In the first experiment we compare Gibbs sampling (*Gibbs*), sampling using local regions (*region*) and sampling using control variables (*control*) in standard regression problems of varied input dimensions. The performance of the algorithms can be accurately assessed by computing the KL divergences between the exact Gaussian posterior $p(\mathbf{f}|\mathbf{y})$ and the Gaussians obtained by Monte Carlo. We fix the number of training points to $N = 200$ and we vary the input dimension d from 1 to 10. Thus we can study the behavior of the algorithms with respect to the amount of correlation in the posterior GP process which depends on how densely the function is discretized. The larger the dimension d, the sparser the discretization of the function is. For each d the training inputs X were chosen randomly inside the unit hypercube $[0, 1]^d$. The outputs Y were chosen by randomly producing a GP function using the squared-exponential kernel $\sigma_f^2 \exp(-\frac{||\mathbf{x}_m - \mathbf{x}_n||^2}{2\ell^2})$, where $(\sigma_f^2, \ell^2) = (1, 100)$ and then adding noise with variance $\sigma^2 = 0.09$. The number of sampling iterations for all algorithms were chosen to be 3×10^4. We use only 3000 samples (by keeping one sample every 10 iterations) to calculate the means and covariances of the 200-dimensional posterior Gaussians. The burn-in period where set to 10^4 iterations[4]. For a certain dimension d the algorithms were initialized to the same state obtained by randomly drawing from the GP prior. The parameters $(\sigma_f^2, \ell^2, \sigma^2)$ were fixed to the values that generated the data. The experimental setup was repeated 10 times so as to obtain confidence intervals. Figure 3 shows the evolution of the KL divergences with respect to the input dimension. Clearly *Gibbs* is very inefficient in low dimensions because of the highly correlated posterior GP process. As d increases and the functions become sparsely discretized, *Gibbs* improves and eventually drops the KL divergences close to zero. The *region* algorithm works better than *Gibbs* but in low dimensions it suffers also from the problem of high correlation. The *control* algorithm makes the KL divergences very close to zero for all dimensions. Note also that as we increase the number of dimensions *Gibbs* eventually becomes slightly better than the *control* algorithm (for $d = 8$ and onwards) since the function values tend to be independent from one another. As shown in Figure 3c, the number of required control variables increases with the dimension. This is very intuitive, since in the limit when the training function values become independent there will not be a sensible low-dimensional representation of the function values and thus we may have to use as many control variables as the number of training function values.

In the next two experiments we apply the *control* algorithm to infer the protein concentration of TFs that active or repress a set of target genes. The latent function in these problems is always one-dimensional and densely discretized and thus the *control* algorithm is the only one that can converge to the GP posterior process in a reasonable time.

In the first experiment we consider the TF p53 which is a tumour repressor activated during DNA damage. According to [2], irradiation is performed to disrupt the equilibrium of the p53 network and the transcription of p53 target genes are then stimulated. Seven samples of the expression levels of the target genes in three replicas are collected as the raw time course data. The non-linear activation of the protein follows the Michaelis Menten kinetics inspired response [1] that allows saturation effects to be taken into account so as $g(f(t)) = \frac{f(t)}{\gamma_j + f(t)}$ in eq. (7) where the Michaelis constant for the jth gene is given by γ_j. Note that since $f(t)$ is positive the GP prior is place on the $\log f(t)$. To apply MCMC we discretize $\mathbf{f}$ using a grid of $P = 121$ points. During sampling, 7 control variables were needed to obtain the desirable acceptance rate. Running time was 4 hours for 5×10^5 sampling

[4]For *Gibbs* we used 2×10^4 iterations since the *region* and *control* algorithms require additional iterations during the adaption phase.



(a) (b) (c)

Figure 3: (a) and (b) show the mean values (with one-standard error bars) of the two KL divergences between the true posterior and the empirically estimated posteriors obtained by *Gibbs*, *region* and *control* algorithms. (c) show the number of regions used by the *region* algorithm and the number of control variables used by the *control* algorithm with respect to the dimension.

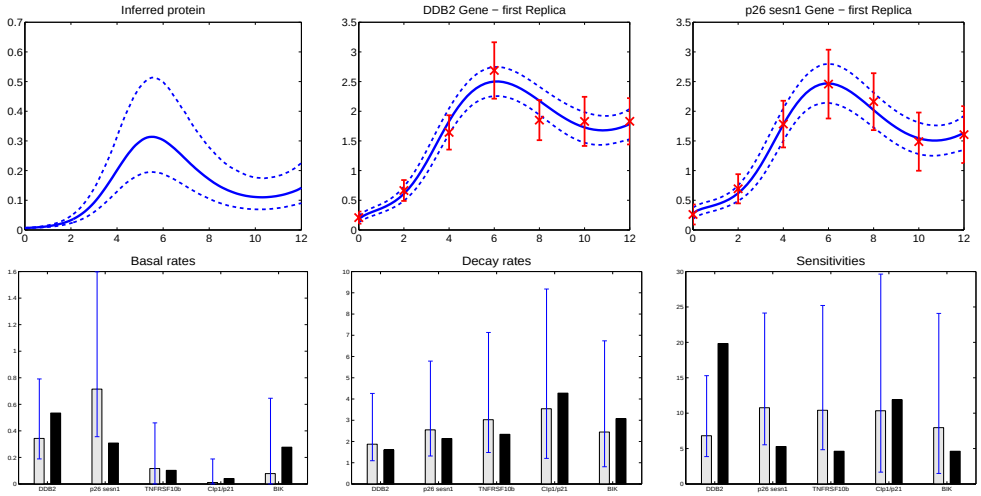


Figure 4: The first row shows the inferred TF (left) and the predicted expressions of two different genes by the ODE model. Red crosses correspond to the actual gene expression measurements. The second row shows the estimated kinetic parameters of the 5 target genes. With grey bars we display the parameters found by our MCMC algorithm and with black bars the parameters found in [2] using a linear ODE model. Error bars correspond to 95% confidence intervals obtained using percentiles.

iterations plus 5×10^4 burn-in iterations. Figure 4 summarizes the estimated quantities obtained from MCMC simulation.

In the second experiment we consider the TF LexA in E.Coli that acts as a repressor. In the repression case there is an analogous Michaelis Menten model [1] where the non-linear function g takes the form: $g(f(t)) = \frac{1}{\gamma_j + f(t)}$. Again the GP prior is placed on the log of the TF activity. We applied our method to the same microarray data considered in [12] where mRNA measurements of 14 target genes are collected over six time points. The amount of LexA is reduced after UV irradiation, decreasing for a few minutes and then recovering to its normal level. For this dataset, the expression of the 14 genes were available for $T = 6$ times. The GP function $\mathbf{f}$ was discretized using 121 points and the *control* algorithm uses 6 control variables. The result for the inferred TF profile along with predictions of two target genes are shown in Figure 5. Our inferred TF profile and reconstructed target gene profiles are similar to those obtained in [12]. However, for certain genes, our model provides a better fit to the gene profile. Additionally, the MCMC approach gives an overall better fit to the gene profiles compared to a Laplace approximation [4].

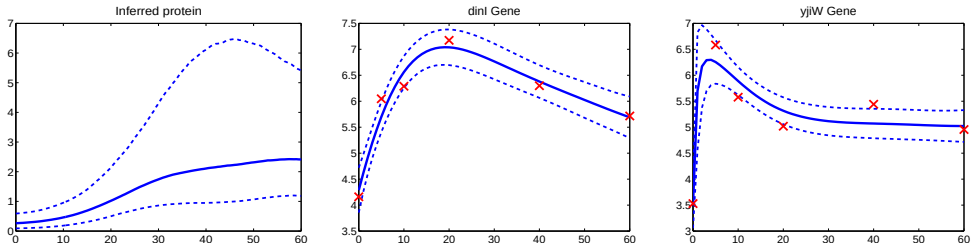


Figure 5: The first plot (left) shows the inferred TF profile, while the predicted expressions of two target genes are shown in the remaining two plots.

6 Discussion

Gaussian processes allow for inference over latent functions using a Bayesian estimation framework. In this paper, we discussed MCMC algorithms that sample functions in GP models. We showed that sampling using control variables can efficiently deal with highly correlated posterior GP processes. MCMC allows for full Bayesian inference in the transcription factor networks application. An important direction for future research will be scaling the models used to much larger systems of ODEs with multiple interacting transcription factors. The algorithm using control variables can be useful for other Gaussian process applications such as those that arise in geostatistics and spatio-temporal models.

References

[1] U. Alon. *An Introduction to Systems Biology: Design Principles of Biological Circuits*. Chapman and Hall/CRC, 2006.

[2] M. Barenco, D. Tomescu, D. Brewer, J. Callard, R. Stark, and M. Hubank. Ranked prediction of p53 targets using hidden variable dynamic modeling. *Genome Biology*, 7(3), 2006.

[3] L. Csato and M. Opper. Sparse online Gaussian processes. *Neural Computation*, 14:641–668, 2002.

[4] P. Gao, A. Honkela, N. Lawrence, and M. Rattray. Gaussian Process Modelling of Latent Chemical Species: Applications to Inferring Transcription Factor Activities . In *ECCB08, to appear*, 2008.

[5] A. Gelman, J. Carlin, H. Stern, and D. Rubin. *Bayesian Data Analysis*. Chapman and Hall, 2004.

[6] M. N. Gibbs and D. J. C. MacKay. Variational Gaussian process classifiers. *IEEE Transactions on Neural Networks*, 11(6):1458–1464, 2000.

[7] N. D. Lawrence, G. Sanguinetti, and M. Rattray. Modelling transcriptional regulation using gaussian processes. In B. Scholkopf, J. C. Platt, and T. Hofmann, editors, *Advances in Neural Information Processing Systems, 19*. MIT Press, 2007.

[8] R. M. Neal. Monte Carlo implementation of Gaussian process models for Bayesian regression and classification. Technical report, Dept. of Statistics, University of Toronto, 1997.

[9] J. Quinonero Candela and C. E. Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005.

[10] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[11] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer-Verlag, 2nd edition, 2004.

[12] S. Rogers, R. Khanin, and M. Girolami. Bayesian model-based inference of transcription factor activity. *BMC Bioinformatics*, 8(2), 2006.

[13] M. Seeger, C. K. I. Williams, and N. D. Lawerence. Fast forward selection to speed up sparse Gaussian process regression. In *In C.M. Bishop and B. J. Frey, editors, Proceedings of the Ninth International Workshop on Aritficial Intelligence*. MIT Press, 2003.

[14] E. Snelson and Z. Ghahramani. Sparse Gaussian process using pseudo inputs. In *Advances in Neural Information Processing Systems, 13*. MIT Press, 2006.

[15] V. Vyshemirsky and M. Girolami. Bayesian Ranking of Biochemical System Models. *Bioinformatics*, 24(6):833–839, 2008.

[16] C. K. I. Williams and D. Barber. Bayesian classification with Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1342–1351, 1998.

Two problems with variational expectation maximisation for time-series models

Richard E. Turner, Pietro Berkes, and Maneesh Sahani
Gatsby Computational Neuroscience Unit
17 Alexandra House, Queen Square, London, WC1N 3AR, London
{turner, berkes, maneesh}@gatsby.ucl.ac.uk

Abstract

Variational methods are a key component of the approximate inference and learning toolbox. These methods fill an important middle ground, retaining distributional information about uncertainty in latent variables, unlike *maximum a posteriori* methods (MAP), and yet requiring fewer computational resources than Monte Carlo Markov Chain methods. In particular the variational Expectation Maximisation (vEM) and variational Bayes algorithms, both involving variational optimisation of a free energy, are widely used in time-series modelling. Here, we investigate the success of vEM in simple probabilistic time-series models. First we consider the inference step of vEM, and show that a consequence of the well-known compactness property is a failure to propagate uncertainty in time, thus limiting the usefulness of the retained distributional information. In particular, the uncertainty may appear to be smallest precisely when the approximation is poorest. Second, we consider parameter learning and analytically reveal systematic biases in the parameters found by vEM. Surprisingly, simpler variational approximations (such a mean-field) can lead to less bias than more complicated structured approximations.

1 The variational approach

We begin with a very brief review of vEM. The Expectation-Maximisation (EM) algorithm [1] is a standard approach to finding maximum likelihood (ML) parameters for latent variable models, including hidden Markov Models and linear or non-linear state space models (SSMs) for time-series. The algorithm can be re-formulated as a variational optimisation of a free-energy [2, 3]. Consider observations collected into a set Y, that depend on latent variables X and parameters θ. We seek to maximise $\log p(Y|\theta)$ with respect to θ. By introducing a new distribution over the latent variables $q(X)$, we can write

$$\log p(Y|\theta) = \log \int dX\ p(Y,X|\theta) = \log \int dX\ p(Y,X|\theta)\frac{q(X)}{q(X)}, \tag{1}$$

$$\geq \int dX\ q(X) \log \frac{p(Y,X|\theta)}{q(X)} = F(q(X),\theta). \tag{2}$$

This last quantity is the free energy. It is smaller than the log-likelihood by an amount equal to the Kullback-Leibler (KL) divergence between $q(X)$ and the posterior distribution on the latents $p(X|Y,\theta)$

$$F(q(X),\theta) = \log p(Y|\theta) - \mathrm{KL}(q(X)||p(X|Y,\theta)), \tag{3}$$

For fixed θ, the optimum value for q is clearly equal to $p(X|Y,\theta)$, at which point the KL divergence vanishes and the free energy equals the log-likelihood. Thus, alternate maximisation of $F(q,\theta)$ with respect to q (the E-step) and θ (the M-step) will eventually find parameters that maximise the likelihood.

In many models, calculation of this posterior is intractable. Thus, the vEM approach is to instead optimise q restricted to a class of distributions $\mathcal{Q}$, within which the minimum of the KL divergence

can tractably be found. The optimal q is called the variational approximation to the posterior. Constrained optimisation of q now alternates with optimisation of θ to find a maximum of the free energy, though not necessarily the likelihood. The optimal parameters are taken to approximate the ML values.

Most often, the class $\mathcal{Q}$ is defined to contain all distributions that factor over disjoint sets C_i of the latent variables in the problem: $q(X) = \prod_{i=1}^{I} q_i(x_{C_i})$. For example, if each latent variable appears in a factor of it own, the approximation is called *mean-field*. Partial factorisations, which keep some of the dependencies between variables are called *structured approximations*. In both cases the q_i's are found iteratively, by repeating the following updates,

$$q(x_i) \propto \exp\left(\langle \log p(Y, X|\theta)\rangle_{\prod_{j\neq i} q_j(x_{C_j})}\right). \tag{4}$$

Here, we analyse the accuracy of vEM in two stages. We first look at the relationship between the true posterior distribution and the variational approximation. It is well known that variational methods tend to be compact [4]. For instance, a unimodal variational approximation to a multimodal distribution will match the largest mode [5], rather than averaging across all of them, and a spherical Gaussian variational approximation will match the shortest length-scale of a correlated Gaussian. We show that this compactness results in a complete failure to propagate uncertainty between time-steps, often making the variational approximation most over-confident exactly when it is poorest. We then consider the accuracy of the vEM parameter estimates. As the variational bound on the likelihood is parameter dependent, variational methods can be biased away from peaks in the likelihood, toward regimes where the bound in tighter. As a result, the best approximations for learning are not necessarily the tightest, but rather those that result in bounds which depend least on the parameters. Both of these properties are exemplified using simple time-series models, although the conclusions are likely to apply more generally.

2 Variational approximations do not propagate uncertainty

Fully factored variational approximations (so called mean-field approximations) have been used for inference in time-series models as they are fast and yet still return estimates of uncertainty in the latent variables [6]. Here, we show that in a simple model, the variational iterations fail to propagate uncertainty between the factors, rendering these estimates of uncertainty particularly inaccurate in time-series (see [7] for a related example).

We consider a time-series model with a single latent variable x_t at each time-step drawn from an AR(1) prior with coefficient λ and innovations variance σ^2,

$$p(x_t|x_{t-1}) = \mathrm{Norm}(\lambda x_{t-1}, \sigma^2). \tag{5}$$

The marginal mean of this distribution is zero and the marginal variance is $\sigma_\infty^2 = \frac{\sigma^2}{1-\lambda^2}$. Typically the latent variables are assumed carry strong temporal correlations, so that λ is close to 1[1]. We consider arbitrary instantaneous likelihood functions, $p(y_t|x_t)$. Using an approximating distribution which is factored over time $q(x_{1:T}) = \prod_{t=1}^{T} q(x_t)$, the update for the latent variable at time t follows from Eq. 4,

$$q(x_t) = \frac{1}{Z} p(y_t|x_t) \exp(\langle \log p(x_t|x_{t-1}) p(x_{t+1}|x_t)\rangle_{q(x_{t-1})q(x_{t+1})}), \tag{6}$$

$$= \frac{1}{Z'} p(y_t|x_t) \mathrm{Norm}\left(\frac{\lambda}{1+\lambda^2}\left(\langle x_{t-1}\rangle + \langle x_{t+1}\rangle\right), \frac{\sigma^2}{1+\lambda^2}\right) = \frac{1}{Z'} p(y_t|x_t) q_{\mathrm{prior}}(x_t). \tag{7}$$

That is, the variational update is formed by combining the likelihood with a variational prior-predictive $q_{\mathrm{prior}}(x_t)$ that contains the contributions from the latent variables at the adjacent time-steps. This variational prior-predictive is interesting because it is identical to the true prior-predictive when there is no uncertainty in the adjacent variables. That is, *none* of the (potentially large) uncertainty in the value of the adjacent latent variables is propagated to $q(x_t)$, and the width of the varia-

[1] In fact the effective time-scale of Eq.5 is $\tau_{eff} = -1/\log(\lambda)$ and so a change in λ from 0.9 to 0.99 is roughly equivalent to a change from 0.99 to 0.999. This is important when the size of the biases in the estimation of λ are considered.

tional predictive is consequently narrower than the width of state-conditional distribution $p(x_t|x_{t-1})$ (compare to Eq. 5)[2].

Temporally factored variational methods for time-series models will thus generally recover an approximation to the posterior which is narrower than the state-conditional distribution. As the whole point of time-series models is that there are meaningful dependencies in the latents, and therefore the state-conditional often has a small width, the variational uncertainties may be tiny compared to the true marginal probabilities. Thus, the mean-field approach essentially reduces to iterative MAP-like inference, except that we find the mean of the posterior rather than a mode. In the next section, it will be shown that this does have some advantages over the MAP approach, notably that pathological spikes in the likelihood can be avoided.

In conclusion, although variational methods appear to retain some information about uncertainty, they fail to propagate this information between variables. In particular, in time-series with strong correlations between latents at adjacent times, the variational posterior becomes extremely concentrated, even though it is least accurate. An ideal distributional approximation would perhaps behave in the opposite fashion, returning larger uncertainty when it is likely to be more inaccurate.

3 Variational approximations are biased

In the last section we showed that variational approximations under-estimate the uncertainties in inference. We now ask how these inaccuracies might affect the parameter estimates returned by vEM. This question is important in many contexts. For example, scientific enquiry is often concerned with the values of a parameter, to substantiate claims like "natural scenes vary slowly" or "natural sounds are sparse", for instance.

What makes for a good variational approximation in this case? The instant reaction is that the free-energy should be as close to the likelihood as possible. That is $\mathrm{KL}(q(X)||p(X|Y,\theta))$ should be as small as possible for all X. However, from the perspective of learning it is more important to be *equally tight everywhere*, or in other words it is more important for the KL-term to be as parameter-independent as possible: If $\mathrm{KL}(q(X)|p(X|Y,\theta))$ varies strongly as a function of the parameters, this can shift the peaks in the free-energy away from the peaks in the likelihood, toward the regions were the bound is tighter. (See [8] for a related example for variational Bayes in mixture models.)

We now illustrate this effect in a linear SSM. In particular, we show that the mean-field approximation can actually have less severe parameter-dependent biases than two structural approximations, and can therefore lead to better vEM parameter estimates, even though it is less tight everywhere.

Deriving the learning algorithms
In the following we first introduce an elementary SSM, for which we can find the exact likelihood ($\log p(y|\theta)$). We then examine the properties of a set of different variational learning algorithms. This set comprises a mean-field approximation, two different structural approximations, and zero-temperature EM. This final approximation can be thought of as vEM where the approximating distributions are delta functions centred on the *maximum a posteriori* (MAP) estimates [3]. The analysis of these schemes proceeds as follows: First the optimal E-Step updates for these approximations are derived; Second, it is shown that, as the SSM is a simple one, the free-energies and the zero-temperature EM objective function can be written purely in terms of the parameters. That is, $\max_{q(x)} F(\theta, q(x))$ and $\max_X \log p(Y, X|\theta)$ have closed form solutions, and do not require iterative updates to be computed as is usual. Thus, we can study the relationship between the peaks in the likelihood and the peaks in the free-energies and zero-temperature EM objective function, for any dataset. An outline of the derivation of these quantities is given here, but for more detail see the associated technical report [9].

Consider an SSM which has two latent variables per time-step and two time-steps. We take the priors on the latent variables to be linear-Gaussian, and the observations are given by summing the

[2]This problem only gets worse if the prior dynamics have longer dependencies, e.g. if $p(x_t|x_{t-1:t-\tau}) = \mathrm{Norm}(\sum_{t'=1}^{\tau} \lambda_{t'} x_{t-t'}, \sigma^2)$ then the variational prior-predictive has a variance, $\frac{\sigma^2}{1+\sum_{t'=1}^{\tau} \lambda_{t'}^2}$.

latents at the corresponding time-step and adding Gaussian noise,

$$p(x_{k,1}) = \mathrm{Norm}\left(0, \frac{\sigma_x^2}{1-\lambda^2}\right), \tag{8}$$

$$p(x_{k,2}|x_{k,1}) = \mathrm{Norm}\left(\lambda x_{k,1}, \sigma_x^2\right), \tag{9}$$

$$p(y_t|x_{1,t}, x_{2,t}) = \mathrm{Norm}(x_{1t} + x_{2t}, \sigma_y^2). \tag{10}$$

This defines a joint Gaussian over the observations and latent variables. From this we can compute the likelihood exactly by marginalising,

$$p(y_1, y_2|\theta) = \mathrm{Norm}(0, \Sigma_Y), \quad \Sigma_Y = I\sigma_y^2 + 2\frac{\sigma_x^2}{1-\lambda^2}\begin{bmatrix} 1 & \lambda \\ \lambda & 1 \end{bmatrix}. \tag{11}$$

The posterior distribution over the latent variables is also Gaussian, and is given by, $p(\mathbf{x}|\mathrm{y}) = \mathrm{Norm}(\mu_{\mathbf{x}|\mathrm{y}}, \Sigma_{\mathbf{x}|\mathrm{y}})$, where $\mathbf{x} = [x_{11}, x_{21}, x_{12}, x_{22}]^T$. The covariance and mean are

$$\Sigma^{-1}_{\mathbf{x}|\mathrm{y}} = \begin{bmatrix} \frac{1}{\sigma_y^2} + \frac{1}{\sigma_x^2} & \frac{1}{\sigma_y^2} & -\frac{\lambda}{\sigma_x^2} & 0 \\ \frac{1}{\sigma_y^2} & \frac{1}{\sigma_y^2} + \frac{1}{\sigma_x^2} & 0 & -\frac{\lambda}{\sigma_x^2} \\ -\frac{\lambda}{\sigma_x^2} & 0 & \frac{1}{\sigma_y^2} + \frac{1}{\sigma_x^2} & \frac{1}{\sigma_y^2} \\ 0 & -\frac{\lambda}{\sigma_x^2} & \frac{1}{\sigma_y^2} & \frac{1}{\sigma_y^2} + \frac{1}{\sigma_x^2} \end{bmatrix}, \quad \mu_{\mathbf{x}|\mathrm{y}} = \frac{1}{\sigma_y^2}\Sigma_{\mathbf{x}|\mathrm{y}}\begin{bmatrix} y_1 \\ y_1 \\ y_2 \\ y_2 \end{bmatrix}. \tag{12}$$

The posterior is correlated through time because of the linear-Gaussian prior, and correlated across chains because of explaining away. The correlations through time increase as the prior becomes slower ($|\lambda|$ increases) and less noisy (σ_x^2 decreases). The correlations across chains increase as the observation noise (σ_y^2) decreases.

We now derive the optimal E-Step for four different approximations: The first three approximations provide uncertainty estimates and these are the fully factored mean-field approximation (q_1), factorisation over chains but not time (q_2), and factorisation over time but not chains (q_3), as shown in the following table: The optimal E-Step updates for these three distributions can be found by

	factored over time	unfactored over time
factored over chains	$q_1(\mathbf{x}) = q_{11}(x_{11})q_{12}(x_{12})q_{13}(x_{21})q_{14}(x_{22})$	$q_2(\mathbf{x}) = q_{21}(x_{11}, x_{12})q_{22}(x_{21}, x_{22})$
unfactored over chains	$q_3(\mathbf{x}) = q_{31}(x_{11}, x_{21})q_{32}(x_{12}, x_{22})$	$p(\mathbf{x}|\mathrm{y}) = q(x_{11}, x_{12}, x_{21}, x_{22})$

minimising the variational KL. Each factor is found to be Gaussian, with a mean and precision that match the corresponding elements in $\mu_{\mathbf{x}|\mathrm{y}}$ and $\Sigma^{-1}_{\mathbf{x}|\mathrm{y}}$. The fourth and final approximation is zero-temperature EM (q_4), for which the E-Step is given by the MAP estimate for the latent variables. As the posterior is Gaussian, the mode and the mean are identical and so the MAP estimates are identical to the variational values for the means.

The next step is to compute the free-energies. In the first three cases, the Gaussianity of the posterior as well as q_1, q_2, and q_3 makes it possible to compute the KL divergences analytically:

$$\mathrm{KL}_i\left(\prod_{a=1}^{A} q_{ia}(\mathbf{x}_a)||p(\mathbf{x}|\mathrm{y})\right) = \frac{1}{2}\log\frac{\prod\Sigma_{ia}}{\Sigma_{\mathbf{x}|\mathrm{y}}}. \tag{13}$$

Using this expression we find,

$$\mathrm{KL}_1 = \frac{1}{2}\log\frac{\left(\sigma_y^2 + \sigma_x^2\right)^4}{\sigma_y^4\gamma}, \ \mathrm{KL}_2 = \frac{1}{2}\log\frac{\left(\left(\sigma_y^2 + \sigma_x^2\right)^2 - \lambda^2\sigma_y^4\right)^2}{\sigma_y^4\gamma}, \tag{14}$$

$$\text{and } \mathrm{KL}_3 = \frac{1}{2}\log\frac{\left(\sigma_y^2 + 2\sigma_x^2\right)^2}{\gamma}, \tag{15}$$

where $\gamma = (1-\lambda^2)\left((2\sigma_x^2 + \sigma_y^2)^2 - \lambda\sigma_y^4\right)$. In the fourth approximation, the KL divergence between a Gaussian and a delta function is infinite. Therefore, the KL term is discarded for zero-temperature EM and the log-joint is used as a pseudo-free energy.

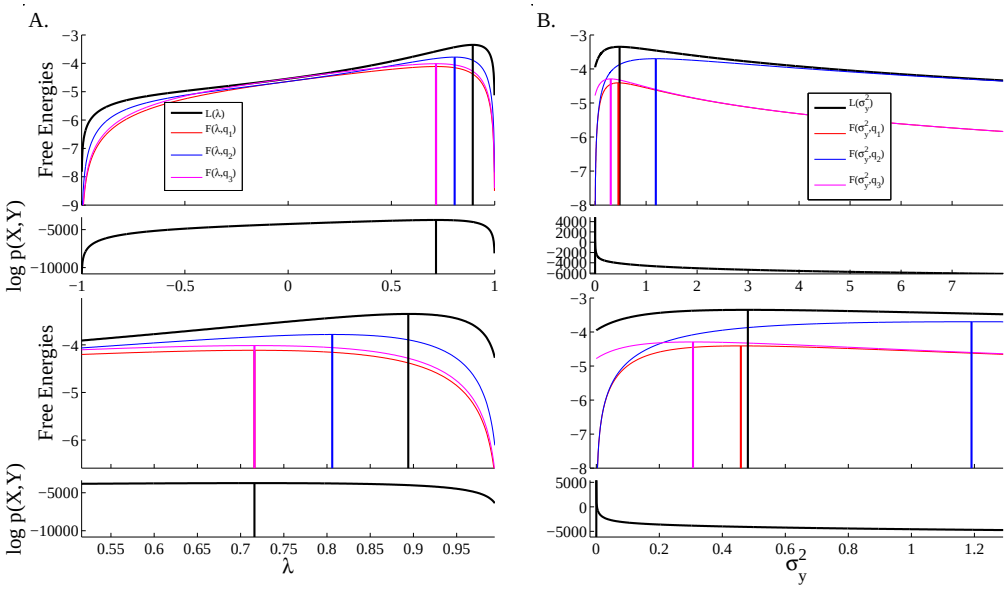


Figure 1: Biases in the Free-energies for a simple linear dynamical system. True/ML parameters are $\lambda = 0.9$, $\sigma_x^2 = 1 - \lambda^2 = 0.19$, and $\sigma_y^2 = 0.43$. In each case one parameter is learned and the others are set to their true/ML values. A. learning λ, B. learning σ_y^2. Large panels show the uncertainty preserving methods ($q_{1:3}$). Small panels show the zero-temperature EM approach (q_4). The bottom two panels show a zoomed in region of the top two panels.

General properties of the bounds: A sanity check

We now verify that these results match our intuitions. For example, as the mean field approximation is a subclass of the other approximations, it is *always* the loosest of the bounds, $\mathrm{KL}_1 > \mathrm{KL}_2, \mathrm{KL}_3 > 0$. Furthermore, approximation 3 (factorising over time) becomes looser than approximation 2 (which does not) when temporal correlations dominate over the correlations between chains. This is indeed the case as $\mathrm{KL}_3 > \mathrm{KL}_2$ when $r = \frac{\sigma_x^2}{|\lambda|\sigma_y^2} < 1$. Moreover, approximation 2 (which factorises over chains) is equivalent to the mean field approximation, $\mathrm{KL}_1 = \mathrm{KL}_2$, when there are no temporal correlations, $\lambda = 0$ or $\sigma_x^2 = \infty$, and in this case the true posterior matches approximation 3, $\mathrm{KL}_3 = 0$. Similarly, approximation 3 is equivalent to the mean-field approximation when the observation noise is infinity $\sigma_y^2 = \infty$, and here approximation 2 is exact $\mathrm{KL}_2 = 0$.

We can now consider how the maxima in the likelihood relate to the maxima in the Free-energies. Unfortunately, there is no closed form solution for these maxima, but in the simple examples which follow, the free-energies and likelihoods can be visualised. In general, we use as our data-set a large number of samples drawn from the forward model ($N > 10000$) and in all cases the ML parameters are essentially equal to the true parameters.

The model has a total of three parameters. We first consider learning just one of these parameters and set the others to the true/ML value. This will allow us to develop some intuition about the ways in which different approximations lead to different biases in the parameter estimates. In this case, the likelihood and free-energies are easy to visualise; some typical examples are shown in Fig. 1. We then consider how the bias changes as a function of the true/ML parameters, and observe that there is no universally preferred approximation, but instead the least biased approximation depends on the parameter that is being learned and on the value of the true/ML parameters. Finally, in we will study the bias when learning the dynamic parameter and the observation noise simultaneously.

Learning the dynamical parameter, λ

We begin by considering learning λ, with the other parameters fixed. As the magnitude of the dynamical parameter increases, so does the correlation in the posterior between successive latent variables in the same chain, that is $x_{k,1}$ and $x_{k,2}$. This means the factorisation over time results in looser bounds as the magnitude of λ increases (KL_3 increases, Eq. 3). Furthermore, as the

correlation between latents in the same chain increases, ($x_{k,1}$ and $x_{k,2}$), so does the correlation between x_{11} and x_{22} (propagated by the explaining away). This means, somewhat surprisingly, that the approximation which does not factorise over time, but over chains, also becomes looser as the magnitude of λ increases. That is, KL_2 increases with the magnitude of λ. Due to the fact that both bounds become less tight as λ increases, the free-energies peak at lower values of λ than the likelihood does, and therefore yield under-estimates (see [10] for a similar result).

The mean-field approximation suffers from both of the aforementioned effects, and it is therefore looser than both. However, with regard to their dependence on λ, KL_1 and KL_3 are equivalent. This means that the mean field approximation and the approximation that factors over time recover identical values for the dynamical parameter, even though the former is looser. Curiously, the solution from zero-temperature EM is also *identical to the mean-field (q_1) and temporally factored (q_3) solutions*. One of the conclusions to draw from this is that most severe approximation need not necessarily yield the most biased parameter estimates.

Learning the observation noise, σ_y^2, and the dynamical noise, σ_x^2

Next we consider learning σ_y^2, with the other parameters fixed to their true values. Due to explaining away, decreasing the observation noise increases the correlation between variables at the same time step, i.e., between x_{1t} and x_{2t}. This means that the approximation that factors over chains, becomes worse as σ_y^2 decreases, and therefore KL_2 is an increasing function of σ_y^2 . In contrast, the approximation that factorises over time, but not over chains, becomes tighter as σ_y^2 decreases i.e. KL_3 is a decreasing function of σ_y^2. As the mean-field approximation shares both of these effects it lies somewhere between the two, depending on the settings of the parameters. This means that whilst approximation 3 under-estimates the observation noise, and approximation 2 over-estimates it, the loosest approximation of the three, the mean field approximation, can actually provide the best estimate, as its peak lies in between the two. The purpose of the next section is to characterise the parameter regime over which this occurs.

In contrast to the situation with the dynamical parameter, the zero-temperature EM objective behaves catastrophically as a function of the observation noise, σ_y^2. This is caused by a narrow spike in the likelihood-surface at $\sigma_y^2 = 0$. At this point the latent variables arrange themselves to explain the data perfectly, and so there is no likelihood penalty (of the sort $-\frac{1}{2\sigma_y^2}(y_t - x_{1,t} - x_{2,t})^2$). In turn, this means the noise variance can be shrunk to zero which maximises the remaining terms ($\propto -\log \sigma_y^2$). The small cost picked up from violating the prior-dynamics is no match for this infinity.

This is not a very useful solution from either the perspective of learning or inference. It is a pathological example of overfitting[3]: There is an infinitesimal region of the likelihood-posterior surface with an infinite peak. By integrating over the latent variables, in a variational method for example, the problem vanishes as the peak has negligible mass and so makes only a small contribution. So, although variational methods often do not preserve as much uncertainty information as we would like, and are often biased, by recovering means and not modes they provide better joint estimates than the catastrophic zero-temperature EM approach.

Learning the dynamical noise σ_x^2 with the other parameters fixed at their true values results in a very similar situation: approximation 2 under-estimates σ_x^2, and approximation 3 over-estimates it, while the mean-field approximation returns a value in between. Once again the MAP solution suffers from an overfitting problem whereby the inferred value of σ_x^2 is driven to zero.

Characterising the space of solutions

In the previous section we found that for a particular setting of the true/ML parameter, the mean-field approximation was the most unbiased (see Fig. 1). How typical is this scenario? One way of answering this question is to evaluate the bias in the parameters learned using the four approximation schemes for many different data-sets each with different maximum-likelihood parameters. In practice three methods are used to find the optimal settings of the parameters. The first is to perform a grid based search, the second is to perform direct gradient ascent on the free-energy and the third is to run vEM. All three methods return identical results up to experimental error.

As a typical example, we show the bias in inferring λ for many different maximum-likelihood settings of σ_y^2 and λ in Fig. 2A. In each case σ_x^2 was set to the ML value, which was close to the

[3]This is the SSM analogue to Mackay's so-called KABOOM! problem in soft K-means [4]

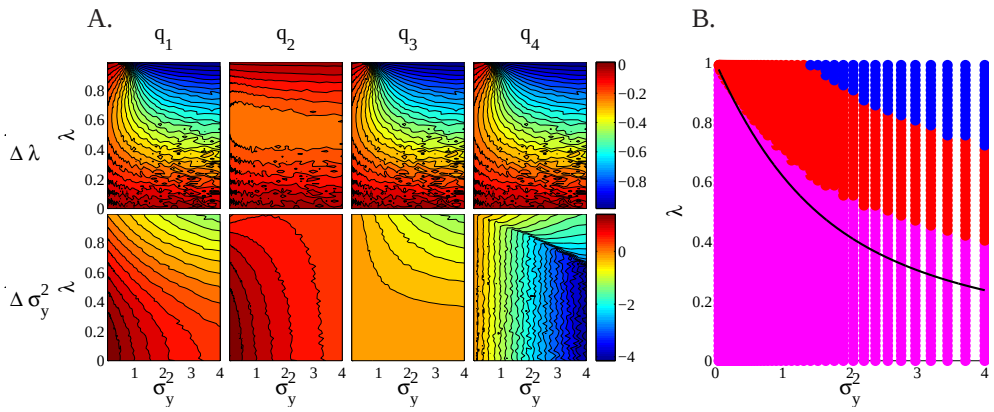


Figure 2: A. Biases for infering a single parameter as a function of σ_y^2 and λ. For all points $\sigma_x^2 = 1 - \lambda^2$. Bias is defined as $\Delta\Theta = \Theta_{INF} - \Theta_{ML}$ so that over-estimation results in a positive bias. Columns correspond to the four approximations. Top Row: Bias in λ. Bottom Row: Bias in σ_y^2.B. The best approximation for finding σ_y^2 indicated by color (q_1 red, q_2 blue and q_3 magenta). The black solid line is $r = \sigma_x^2/|\lambda|\sigma_y^2 = 1$ and below it approximation 3 is tightest, and above it approximation 2 is tightest.

true value of $1 - \lambda^2$. The parameter is under-estimated in all cases, often by a substantial amount (e.g. for approximations 1,3, and 4, at high σ_y^2 and λ values, the bias is almost one). The bias from using approximation 2 is always smaller than that from using the others, and it is to be preferred everywhere. However, this does not generalise for other parameters. The bias for inferring σ_y^2 is shown in Fig. 2B. As noted in the previous section, approximation 2 over-estimates the observation noise, whilst approximation 3 and 4 under-estimate it. The mean-field approximation combines the behaviours of approximation 2 and 3 and therefore under-estimates in regions where λ and σ_y^2 are small, and over-estimates in regions where they are large. In the intermediate region, these effects cancel and this is the region in which the mean-field approximation is the best. This is shown in Fig. 2C which indicates the best approximation to use for inferring the observation noise at different parts of the space. The mean-field solution is to be preferred over a fairly large part of the space.

Which is the best approximation therefore depends not only on which parameter has to be learned, but also on the ML value of parameters.

Simultaneous inference of pairs of parameters

So far we have considered estimating a single parameter keeping the others at their true values. What happens when we infer pairs of parameters at once? Consider, for instance, inferring the dynamical parameter λ and the observation noise σ_y^2 with σ_x^2 held at its ML/true value (see Fig. 3). As before, three methods are used to find the optimal parameter settings (gridding, gradient ascent and vEM). In a small minority of cases the objective functions are multi-modal, in which case the agreement between the methods depends on the initialisation. In order to avoid this ambiguity, the gradient based methods were initialised at the values returned from the method of gridding the space. This procedure located the global optima. The most striking feature of Fig. 3A. is that the biases are often very large (even in regimes where the structural approximations are at their tightest). Moreover, as there is a many to one mapping between the true parameters and the inferred parameters this indicates that it is impossible to simply correct for the variational bias by looking at the inferences.

Fig. 3B. shows that, in contrast to the case where only one parameter is inferred at a time, the mean-field solution is no-longer superior to the structural approximations. It also indicates that whilst tightness is a guide for choosing the best approximation, it is not very accurate. It is also notable that when all three parameters are inferred together (data not shown), the biases become even larger.

Finally, we consider the relevance of this toy example, and in particular what happens in longer time-series ($T > 2$) with more hidden variables ($K > 2$). In general both of these changes result in posterior distributions that have richer correlational structure. (That is, the posterior covariance matrix has more off-diagonal terms.) The variational approximations thus ignore larger parts of this structure and therefore the KL terms and associated biases will become correspondingly larger.

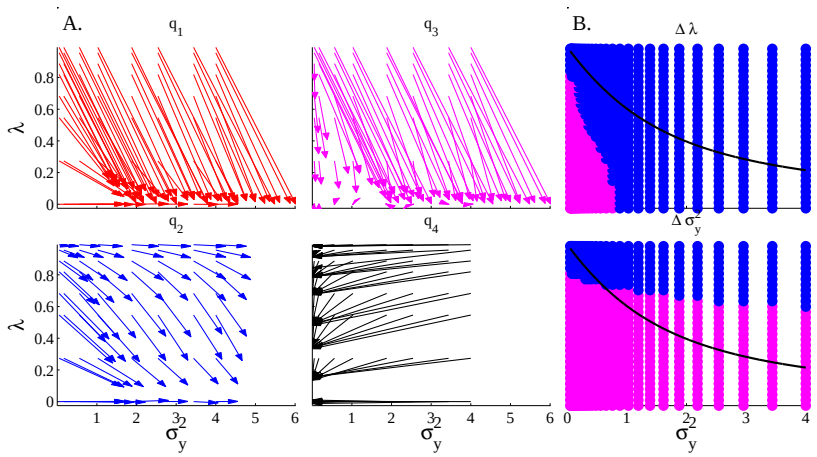


Figure 3: Simultaneous inference of λ and σ_y^2 with biases shown as a function of the true/ML settings of the parameters. A. For each approximation ($q_{1:4}$) a number of simulations are run and each is represented by an arrow. The arrow begins at the true/ML setting of the parameters and the tip ends at the inferred value. Ideally the arrows would be very short, but in fact they are often very large. B. The best uncertainty preserving approximation ($q_{1:3}$) for finding λ (Top) and σ_y^2 (Bottom) indicated by color (q_1 red, q_2 blue and q_3 magenta). The black solid line is $r = \sigma_x^2/|\lambda|\sigma_y^2 = 1$ and below it approximation 3 is tightest, and above it approximation 2 is tightest.

4 Conclusion

We have discussed two problems in the application of vEM to time-series models. First, the compactness property of variational inference leads to a failure to propagate posterior uncertainty through time. Second, the dependence of the variational lower bound on the model parameters often leads to strong biases in parameter estimates. We found that the relative bias of different approximations depended not only on which parameter was sought, but also on its true value. Moreover, tightest bound did not always yield the smallest bias: in some cases, structured approximations were more biased than the mean-field approach. Variational methods did, however, avoid the over fitting problem which plagues MAP estimation. Despite these shortcomings, variational methods remain a valid, efficient alternative to computationally costly Markov Chain Monte Carlo methods. However, the choice of the variational distribution should be complemented with an analysis of the dependency of the variational bound on the model parameters. Hopefully, these examples will inspire new algorithms that pool different variational approximations in order to achieve better performance.

Acknowledgments

We thank David Mackay for inspiration. Supported by the Gatsby Charitable Foundation.

References

[1] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm (with discussion). *J. of the Royal Stat. Society: B*, 39:1–38, 1977.

[2] R.J. Hathaway. Another interpretation of the EM algorithm for mixture distributions. *Statistics & Probability Letters*, 4(2):53–56, 1986.

[3] R.M. Neal and G.E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In Michael I. Jordan, editor, *Learning in Graphical Models*, pages 355–370. Kluwer Academic Press, 1998.

[4] D.J.C. Mackay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003.

[5] Bishop. C.M.M. *Pattern Recognition and Machine Learning*. Springer, 2006.

[6] M.J. Beal. *Variational Algorithms for Approximate Bayesian Inference*. PhD thesis, Gatsby Computational Neuroscience Unit, University College London, 2003.

[7] J. Winn and T. Minka. Expectation propogation and variational message passing: a comparison using infer.net. http://videolectures.net/abi07_winn_ipi/, 2007. NIPS 2007 Workshop Inference in continuous/hybrid models.

[8] D.J.C. Mackay. A problem with variational free energy minimization. 2001.

[9] R.E. Turner and M. Sahani. Failure models of variational methods in toy problems. Technical report, Gatsby Computational Neuroscience Unit, 2008. Report 1.

[10] B. Wang and D.M. Titterington. Lack of consistency of mean field and variational bayes approximations for state space models. *Neural Proc. Lett*, 20(3):151–170, 2004.