Monteiro, André A.

Working Paper

Parameter Driven Multi-state Duration Models: Simulated vs. Approximate Maximum Likelihood Estimation

Tinbergen Institute Discussion Paper, No. 08-021/2

Provided in Cooperation with:
Tinbergen Institute, Amsterdam and Rotterdam

Suggested Citation: Monteiro, André A. (2008) : Parameter Driven Multi-state Duration Models: Simulated vs. Approximate Maximum Likelihood Estimation, Tinbergen Institute Discussion Paper, No. 08-021/2, Tinbergen Institute, Amsterdam and Rotterdam

This Version is available at:
http://hdl.handle.net/10419/86852

TI 2008-021/2
Tinbergen Institute Discussion Paper

Parameter Driven Multi-state Duration Models

André A. Monteiro

UWA Business School, University of Western Australia, Australia, and VU University Amsterdam, The Netherlands, and Tinbergen Institute.

Tinbergen Institute
The Tinbergen Institute is the institute for economic research of the Erasmus Universiteit Rotterdam, Universiteit van Amsterdam, and Vrije Universiteit Amsterdam.

Tinbergen Institute Amsterdam
Roetersstraat 31
1018 WB Amsterdam
The Netherlands
Tel.: +31(0)20 551 3500
Fax: +31(0)20 551 3555

Tinbergen Institute Rotterdam
Burg. Oudlaan 50
3062 PA Rotterdam
The Netherlands
Tel.: +31(0)10 408 8900
Fax: +31(0)10 408 9031

Most TI discussion papers can be downloaded at http://www.tinbergen.nl.

Parameter Driven Multi-state Duration Models: Simulated vs. Approximate Maximum Likelihood Estimation

André A. Monteiro*
Tinbergen Institute, The Netherlands
and University of Western Australia

This version: February 15, 2008

Abstract

Likelihood based inference for multi-state latent factor intensity models is hindered by the fact that exact closed-form expressions for the implied data density are not available. This is a common and well-known problem for most *parameter driven* dynamic econometric models. This paper reviews, adapts and compares three different approaches for solving this problem. For evaluating the likelihood, two of the methods rely on Monte Carlo integration with importance sampling techniques. The third method, in contrast, is based on fully deterministic numerical procedures. A Monte Carlo study is conducted to illustrate the use of each method, and assess its corresponding finite sample performance.

Keywords: Multi-state Duration models, Parameter Driven models, Simulated Maximum Likelihood, Importance Sampling.
JEL classification codes: C15, C32, C33, C41

* Contact details: UWA Business School M251, University of Western Australia, 35 Stirling Highway, Crawley WA 6009, Australia. Telephone: +61864887078, fax: +61864881035, email: AndreAVPBMonteiro@gmail.com. The present paper is part of the Ph.D. thesis the author is doing at the VU University Amsterdam with the supervision of Professor André Lucas.

1 Introduction

Following the pioneering work of Hasbrouck (1991) and Engle and Russell (1998), there has been a significant increase in the development and use of econometric models for analyzing irregularly spaced longitudinal data. Usual applications include empirical studies focusing on the microstructure of financial markets or on the dynamics of corporate credit risk. Examples are Engle and Russell (1998), Bowsher (2007), Engle and Lunde (2003), Bauwens and Veredas (2004), Kavvathas (2000), Lando and Skødeberg (2002), Koopman et al. (2006, 2008) and Duffie et al. (2006, 2007).

Econometric models for point processes are useful in economic contexts in which the timing of the events under study is relevant and the interest is focused at the micro level. It is known that the statistical properties of a time series resulting from the aggregation (or sampling) of one or more empirical (compound) point processes over a fixed-length time grid are strongly dependent on the size of the mesh. An excessively large mesh will hide the characteristics of the mechanism under analysis, while a time unit that is too small will induce artificial volatility in the resulting time series. The problem becomes serious because, in some applications, there are no objective criteria for choosing the appropriate length of the time unit.

As it is the case with models for time series data, statistical models for point processes can be classified as either *observation* or *parameter driven*. Self-exciting point process models like the Autoregressive Conditional Duration (ACD) model (and most derived models) of Engle and Russell (1998), the Autoregressive Conditional Intensity (ACI) model of Russell (1999) or the generalized Hawkes processes of Bowsher (2007), are all observation driven models. Conditional on the observable history of the process, the distribution of future observations is completely specified. Bauwens and Hautsch (2006) and Bauwens and Veredas (2004) have introduced two large classes of parameter driven models for doubly stochastic point processes. These were, respectively, the Stochastic Conditional Intensity (SCI) and the Stochastic Conditional Duration (SCD) models. The dynamic behavior of these models is driven not only by an (appropriate) observable filtration but also by a latent component, thus combining aspects from both self-exciting and doubly stochastic point processes. The richer dynamic structure of this class of models provides added flexibility for describing the patterns in empirical point processes. This added flexibility, however, comes at a cost. Maximum Likelihood (ML) estimation[1] for this class of models is hindered by the need to integrate out the effect of the unobserved component(s). This is a common and well-known problem for parameter driven nonlinear or non-Gaussian

[1]For the SCD model introduced by Bauwens and Veredas (2004) the authors suggest the use of the Kalman filter as a quasi-Maximum Likelihood method.

dynamic statistical models. The data-density typically involves a high-dimensional integral, which has (due to the unavailability of exact closed-form solutions) to be evaluated either using simulation or (other) approximate methods. In this paper I compare the use of two feasible methods for conducting Simulated Maximum Likelihood (SML) estimation and inference, for the Multi-state Latent Factor Intensity Model (introduced in Koopman et al. 2008), against the approximate (numerical) ML method of Davis and Rodriguez-Yam (2005). The two SML methods I consider are the Efficient Importance Sampling (EIS) algorithm of Richard and Zhang (2007) and Liesenfeld and Richard (2003) and the method of Monte Carlo maximum likelihood of Durbin and Koopman (1997, 2000).

The Multi-State Latent Factor Intensity (MLFI) model is a self-exciting and doubly stochastic extension of a continuous-time finite state Markov process (also termed a *Markov chain* in the literature). The generator matrix giving the instantaneous transition rates between the finite set of states is not considered a deterministic function of time, but instead, a (matrix-valued) stochastic process adapted to an appropriate filtration and modulated by latent Gaussian dynamic processes. The MLFI model is a powerful tool for analyzing credit rating data, as shown in Koopman et al. (2006, 2008). Additionally, this statistical model is of potential interest to other fields as well. In fact, the MLFI model is interpretable as a dynamic *frailty* multi-state duration model. Therefore, it is potentially of interest to those scientific areas where frailty models are extensively used (examples include Biostatistics and Labour Economics).

This paper is organized as follows. Section 2 reviews the general MLFI specification. Section 3 describes the two distinct SML methods. In subsection 3.1, the estimation procedure for the MLFI model as described in Koopman et al. (2008) is briefly reviewed. The application of the EIS method of Liesenfeld and Richard (2003) and Richard and Zhang (2007) to the MLFI model is described in subsection 3.2. The approximate ML method proposed in Davis and Rodriguez-Yam (2005) is the focus of section 4. Section 5 contains the details and results of the simulation study. Section 6 concludes.

2 The class of Multi-State Latent Factor Intensity models

Let $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}, \mathbb{P})$ be a given filtered probability space satisfying the usual conditions,[2] with t denoting chronological time. At the heart of every dynamic statistical model with unobserved components lies the decomposition $\mathcal{F}_t = \mathcal{F}_t^o \cup \mathcal{F}_t^*$, where $\mathcal{F}_t^o$ is the information effectively

[2]For a brief discussion see, for example, Andersen et al. (1993)

available to the analyst (therefore including the internal filtration of the stochastic process under consideration), while $\mathcal{F}_t^*$ denotes the set of remaining (both dynamic and time-invariant) factors driving the observable processes. Consider a set of $\tilde{S}$ distinct (right-continuous) counting processes $N_{\tilde{s}}(t)$ defined on $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}, \mathbb{P})$, with $\tilde{s} = 1, \ldots, \tilde{S}$, which are observed over the interval $[0, T]$. Let $\bar{N}_{\tilde{s}}(t)$ denote the corresponding left-continuous counting processes. Assume that the *pooled* counting process $N(t) = \sum_{\tilde{s}=1}^{\tilde{S}} N_{\tilde{s}}(t)$ is *orderly* and the *compensator* $\Lambda_{\tilde{s}}(t)$ associated with $N_{\tilde{s}}(t)$ is absolutely continuous. The corresponding *intensity* process is denoted by $\tilde{\lambda}_{\tilde{s}}(t)$. The MLFI model introduced in Koopman et al. (2008), is obtained by considering that the $\tilde{S}$ distinct counting processes result from recording the transitions experienced by K statistical units as these move across a set of M possible states.[3] Such a multi-state setting is achieved by defining a set of K $\mathcal{F}_t^o$-predictable vector-valued processes $Y^k(t) = (Y_s^k(t))$, with $k = 1, \ldots, K$ and $s = 1, \ldots, S$ denoting the observational unit and transition type respectively $(S = M(M-1))$, as

$$Y_s^k(t) = \begin{cases} 1 & , \text{ unit k is 'at risk' for a transition of type } s \text{ at time } t^- \\ 0 & , \text{ otherwise.} \end{cases} \tag{1}$$

This set of rules is defined on a case-by-case basis, depending on the exact nature of the 'states' in each application. Note that $Y^k(t)$ can be indexed by $\bar{N}^k(t) = \sum_{s=1}^{S} \bar{N}_s^k(t)$, where $\bar{N}_s^k(t)$ is a convenient notation for the individual, left-continuous, counting processes $\bar{N}_{\tilde{s}}(t)$ in the MLFI setting. In the context of credit ratings, a 'state' can, therefore, be interpreted as either a specific credit rating (ex. AAA,AA,A,BBB and so forth in the case of credit ratings from Standard & Poor's) or the combination of the present rating with the information on whether or not that rating was achieved as the result of a downgrade (i.e. the 'state' BBB would be split into the states "downgraded to the rating BBB"and "either BBB as initial rating, or upgraded to BBB").

The intensity process $\tilde{\lambda}_s^k(t) = Y_s^k(t)\lambda_s^k(t)$ for unit k, associated with a transition of type s, can be parameterized, for example, by specifying the conditional hazard rate as a log-linear function of fixed effects, observed covariates and a latent dynamic component,

$$\lambda_s^k(t) = \exp\left[\eta_s + \gamma_s' w^k(t) + \alpha_s \psi^s(t)\right] H_s^k(t). \tag{2}$$

Here η_s is an unknown scalar, $w^k(t)$ is a unit-specific p-dimensional vector containing both constant and time-varying covariates and γ_s is the corresponding vector of coefficients. Component s of the vector $\alpha = (\alpha_1, \ldots, \alpha_S)$ represents the semi-elasticity of the intensity of events of type

[3]Therefore $\tilde{S} = M(M-1)K$ and there are $(M \times (M-1))!$ possible mappings between the set of transition *types* $\mathbb{S} = \{1, 2, \ldots, M(M-1)\}$ and the set of pairs of states $\mathbb{M} = \{(1,2), \ldots, (M, M-1)\}$ involved in any given transition.

s with respect to the latent factor $\psi^s(t)$. The function $H_s^k(t)$ introduces duration dependence in the model, therefore relaxing the Markov assumption. In general,

$$H_s^k(t) = H_s(t - t_0^k \ , \ t - t_1^k \ , \ \dots \ , \ t - t_{N^k(t)}^k),$$

where $t - t_i^k$ denotes the backward-recurrence time (Cox, 1962) of unit k with respect to its past ith transition moment. In this case, the resulting multi-state stochastic process is a generalized semi-Markov one (Glynn, 1988).

In contrast with Koopman et al. (2008), where, in the initial model specification, a single (scalar) latent component influences the different intensity processes, here a more general version of the MLFI model with S independent latent components is described.[4] The vector-valued unobserved stochastic process $\psi(t) = (\psi^1(t), \dots, \psi^S(t))'$, can change values only at the (discrete) set of points $\{t_n\} \in [0, T]$ satisfying the equation

$$\Delta N(t) \equiv N(t) - \bar{N}(t) = 1,$$

where $\bar{N}(t)$ denotes the left-continuous counting process associated with $N(t)$. Therefore, it is possible to index the process ψ using $\bar{N}(t)$. In order to capture latent first order dynamics in the observed transition process, the unobserved vector ψ is specified by the following restricted VAR(1) equation in continuous-time[5]

$$\psi_{\bar{N}(t)+\iota'(t)\iota(t)} = (\rho_1^{\Delta N_1(t)}, \dots, \rho_S^{\Delta N_S(t)})' \odot \psi_{\bar{N}(t)} + \iota(t)\varepsilon_{\bar{N}(t)+1}, \tag{3}$$

where $\iota(t) = (\Delta N_1(t), \dots, \Delta N_S(t))'$ acts as a random selection vector. The autoregressive coefficients ρ_s are restricted to $|\rho_s| \leq 1$ with $s = 1, \dots, S$. The (scalar) disturbances ε_n are an i.i.d. standard Gaussian noise. Although a first order autoregressive specification is used throughout this paper, it is straightforward to extend the dynamics of the latent components, by including in (3) both additional lagged values of ψ and ε.

Over a fixed interval of time, the expected variation of component s of the latent process ψ, as defined in (3), is proportional to the intensity $\lambda_s(t) = \sum_{k=1}^K \lambda_s^k(t)$ of the counting process associated with events of that type. This may not be a desirable property, if, as in Koopman et al. (2006, 2008), the latent process is intended to model a slow moving economy wide risk factor. An alternative specification for ψ that behaves more in line with this interpretation is

$$\psi_{\bar{N}(t)+\iota'(t)\iota(t)} = (\rho_1^{\tau^1_{N_1(t)}\Delta N_1(t)}, \dots, \rho_S^{\tau^S_{N_S(t)}\Delta N_S(t)})' \odot \psi_{\bar{N}(t)} + \sigma_{\bar{N}(t)+1}\iota(t)\varepsilon_{\bar{N}(t)+1}, \tag{4}$$

[4] These are the two extreme cases. In an intermediate situation there can be any number, between 1 and S, of latent components in combination with a particular structure for the vector α of factor loadings.

[5] "$\odot$"denotes Hadamard-Schur (i.e. element-by-element) matrix multiplication.

with

$$\sigma^2_{\bar{N}(t)+1} = \frac{1-(\iota'(t)\rho)^{2\iota'(t)\bar{\tau}_{N(t)}}}{1-(\iota'(t)\rho)^2}.$$

The vector $\rho = (\rho_1, \ldots, \rho_S)'$, stores the (calendar-time) coefficients of mean-reversion, while the vector $\bar{\tau}_n = (\tau^1_{N_1(t_n)}, \ldots, \tau^S_{N_S(t_n)})'$ contains the amount of time elapsed between two consecutive transitions of each type.[6] The disturbances ε_n, again form an i.i.d. standard Gaussian noise process. The specification (4) has small changes in $\psi^s(t)$ over short spells $\tau^s_{N_s(t)}$.

In order to obtain a valid intensity process, the latent innovations ε_n must be assumed independent from the sequence of increments in the compensator associated with the pooled process,

$$\Lambda_n = \sum_{s=1}^{\tilde{S}} \Lambda_s(t_{n-1}, t_n), \quad n = 1, \ldots, N(T),$$

where $\Lambda_s(a,b) = \int_a^b \lambda_s(t)\mathrm{d}t$. A fundamental result, underpinning both the construction of diagnostic tests for intensity based point process models as well as their numerical simulation, is the random time change theorem (see Brown and Nair, 1988, for a proof of this result). This theorem states that every multivariate counting process, whose associated (multivariate) compensator is both absolutely continuous and unbounded, can be mapped into a vector of independent Poisson processes, each with unit intensity, by re-scaling the time axis using the increments in the compensator. In simple terms this implies, in the current context, that

$$\Lambda_n \sim \text{i.i.d. Exp}(1).$$

The model specification made up of equations (2) and either (3) or (4) is incomplete. Changing the sign simultaneously for each α_s and for the complete path of the corresponding component of the vector $\psi(t)$ clearly yields the same path for all intensity processes $\lambda^k_s(t)$ with $k = 1, \ldots, K$. Identification of the MLFI model therefore requires as many sign restrictions as the number of independent components of the latent process ψ. The sign of α_s has to be set a priori (i.e. either $\alpha_s > 0$ or $\alpha_s < 0$).

Specification (2) nests several different models appearing in the literature. For example, when no latent dynamic factors $\psi^s(t)$ are included, and there are neither duration ($H_s(t) \equiv 0$) nor covariate ($\gamma_s \equiv 0$) effects, specification (2) becomes the standard Homogeneous Continuous-Time Markov Chain model used frequently in the Credit Risk literature (Jarrow et al. 1997, Lando and Skødeberg 2002).

As seen in Koopman et al. (2008), for a vector of unknown parameters θ, the likelihood function conditional on the initial ratings, pre-sample event histories, and on the complete path of the

[6] That is, $\tau^s_n = t^s_n - t^s_{n-1}$ and $\{t^s_n\} \subset \mathbb{R}^+_0$ is the set of solutions of the equation $\Delta N_s(t) = 1$.

unobserved process, as defined by $\Psi_{\bar{N}(T)+1} = \{\psi_i\}_{i=1}^{\bar{N}(T)+1}$, can be written as

$$L\left(\theta \mid \mathcal{F}_T^o, \Psi_{\bar{N}(T)}\right) = \prod_{n=1}^{\bar{N}(T)+1} \prod_{k=1}^{K} \prod_{s=1}^{S} \exp\left(\Delta N_s^k(t_n) \ln\{\lambda_s^k(t_n)\} - Y_s^k(t_n) \int_{t_{n-1}}^{t_n} \lambda_s^k(t) \mathrm{d}t\right). \quad (5)$$

In order to estimate the parameter vector θ, the conditional likelihood function must be integrated with respect to the complete path $\Psi_{\bar{N}(T)+1}$ of the unobserved process $\psi(t)$. The maximum likelihood problem becomes

$$\max_{\theta} L(\theta \mid \mathcal{F}_T^o), \quad (6)$$

where

$$L(\theta \mid \mathcal{F}_T^o) = \int L\left(\theta \mid \mathcal{F}_T^o, \Psi_{\bar{N}(T)}\right) p(\Psi_{\bar{N}(T)}) \mathrm{d}\Psi_{\bar{N}(T)}, \quad (7)$$

and $p(\Psi_{\bar{N}(T)})$ denotes the (unconditional) density function of $\Psi_{\bar{N}(T)}$.

3 Simulated Maximum Likelihood estimation

This section starts by recalling the basic concepts of SML estimation and inference using Importance Sampling acceleration techniques. Some basic ideas from state space modelling are also briefly discussed. This provides the adequate background to the description of the Durbin-Koopman methodology in subsection 3.1. This is then followed, in subsection 3.2 by the description of the EIS algorithm.

The realization that many of the relevant factors underlying most economic settings are inherently unobservable, leads to the specification of models with latent random variables. The data density implied by these models, therefore, contains an integral which is interpretable as an expectation taken over the density of the unobserved variables. In general terms, assume that the statistical model specifies, up to an unknown parameter vector θ, the conditional density of the observational vector z given the latent vector x, which in many contexts can be interpreted as a 'structural' density. This conditional density is denoted as

$$p\left(z|x;\theta\right). \quad (8)$$

The unconditional data density then follows from $p\left(z|x;\theta\right)$ and the assumed density for x, denoted by $p\left(x|\theta\right)$

$$p\left(z|\theta\right) = \int p\left(z|x;\theta\right) p\left(x|\theta\right) \mathrm{d}\mu(x) = \mathrm{E}_x\left[p\left(z|x;\theta\right)\right]. \quad (9)$$

The basic idea of simulated maximum likelihood consists in employing Monte Carlo integration to evaluate this integral, by sampling from the density $p\left(x|\theta\right)$, termed the *natural sampler*. In

fact, in the current context, the Weak Law of Large Numbers states that, as $M \uparrow \infty$,

$$M^{-1} \sum_{m=1}^{M} p\left(z|x^m;\theta\right) \xrightarrow{P} \mathrm{E}_x\left[p\left(z|x;\theta\right)\right],$$

where $\{x^m\}$, $m = 1, \ldots, M$ denotes an i.i.d. sample from $p\left(x|\theta\right)$. Accordingly it is possible to perform (approximate) Maximum Likelihood estimation and inference for statistical models with unobserved components using the Monte Carlo estimator

$$\hat{L}_{NS}\left(\theta\right) = M^{-1} \sum_{m=1}^{M} p\left(z|x^m;\theta\right), \tag{10}$$

as a proxy for the exact likelihood function $L\left(\theta\right) = p\left(z|\theta\right)$. The problem with this approach is that the rate of convergence of the estimator in (10) is very slow due to the fact that the latent vector x is sampled directly from the statistical formulation of the model without taking into account the observed data. Therefore for most random draws x^m obtained from the natural sampler, the corresponding contribution to the data likelihood made by $p\left(z|x^m;\theta\right)$ will be very small, thus requiring many random draws. In other words, the direct Monte Carlo estimator given by equation (10) is highly inefficient.

The above reasoning suggests that it may be possible to improve the efficiency of the whole approach by switching from the natural sampler to an auxiliary conditional density $s\left(x|z;\theta_0\right)$ that takes into account the information on x, which is contained in the observed data vector z. This is the central idea of Importance Sampling. Assume such an *auxiliary sampler* is given, the data density implied by the statistical model (9) can then be written as

$$p\left(z|\theta\right) = \int \frac{p\left(z|x;\theta\right) p\left(x|\theta\right)}{s\left(x|z;\theta_0\right)} s\left(x|z;\theta_0\right) \mathrm{d}\mu(x),$$

and use as an estimator of the data likelihood

$$\hat{L}_{IS}\left(\theta\right) = M^{-1} \sum_{m=1}^{M} p\left(z|x^m;\theta\right) \frac{p\left(x^m|\theta\right)}{s\left(x^m|z;\theta_0\right)}, \tag{11}$$

where $\{x^m\}$, $m = 1, \ldots, M$ denotes an i.i.d. sample from the auxiliary density $s\left(x|z;\theta_0\right)$. The critical question is how to construct such an auxiliary density. Obviously the 'ideal' auxiliary sampler is the conditional density of the latent vector given the data, as implied by the model itself, i.e. the 'posterior' density

$$p\left(x|z;\theta\right). \tag{12}$$

In fact if it would be feasible to plug this density, as the auxiliary sampler, inside the estimator (11) this would yield

$$\hat{L}_{IS}\left(\theta\right) = p\left(z|\theta\right),$$

that is, the MC likelihood estimator would become exact (a degenerate random variable equal to the quantity being estimated a.s.). Unfortunately, sampling directly from $p(x|z;\theta)$ is not feasible because there is no analytical expression for this density, by exactly the same reason as for the data density $p(z|\theta)$ itself.[7]

Departing from the same basic idea the two methods described in this section follow, basically, different directions in order to construct an auxiliary sampler that approximates as much as possible the ideal auxiliary sampler $p(x|z;\theta)$. In fact the framework in which both methods are placed is somewhat more specific than what has been discussed so far. Until this point no mention was made regarding the source of the multivariate nature of both the observation vector z and the state vector x. Therefore, both pure cross-sectional, pure time series as well as mixed (i.e. panel data or repeated cross-sections) models with unobserved components were covered by the previous discussion. However, as the main focus of this paper is the estimation of doubly stochastic point process models, I will not mention any methods for dealing with pure cross-sectional models. In fact the three methodologies reviewed in this paper were initially developed in the context of dynamic models. The longitudinal dimension is the main source (although not the only one) of the multidimensionality of both z and x.

3.1 The Durbin-Koopman methodology

The Monte Carlo Maximum Likelihood methodology of Durbin and Koopman (1997) was developed for handling nonlinear and/or non-Gaussian models in state space form. Therefore, this section starts by briefly reviewing the main notions and concepts of (generalized) state space modelling. The main concept is that of the *state* of a dynamic system. The state is defined as the smallest vector, such that, the knowledge of its value at time $t = t_0$ together with knowledge of the path of the input vector for any $t \geq t_0$ determines completely the behaviour of the system at any time $t \geq t_0$.[8].

Denote by $\mathbf{z}_n = (z_1', \ldots, z_n')'$ the (observed) history of the vector process $\mathbf{z}$ up to the present moment t_n (z_i represents the value of this vector at time t_i, $i = 1, \ldots, N$). Similarly, let $\mathbf{x}_n = (x_1', \ldots, x_n')'$ denote the (unobserved) past values of the state vector x. The general non-Gaussian state space model is characterized by the two following properties (subsumed in the

[7] obviously $p(x|z;\theta) = \frac{p(z|x;\theta)p(x|\theta)}{p(z|\theta)}$ and therefore sampling from this density would require having already solved the integral appearing in the data density, which is the source of the whole problem.

[8] This is the classical definition for deterministic dynamic systems. However, random dynamic systems can be thought of as deterministic systems subject to a random input. In this case, "knowledge"of the input vector means knowledge of its probability law. Similarly, the wording "behaviour of the system "refers to the (conditional) density of both the output and state vectors

observation and *state equations*),

$$p(z_n|x_n, \mathbf{x}_{n-1}, \mathbf{z}_{n-1}, \theta) = p(z_n|x_n, \theta) = f(z_n; x_n, \theta), \tag{13}$$

and

$$p(x_{n+1}|x_n, \mathbf{x}_{n-1}, \mathbf{z}_n, \theta) = p(x_{n+1}|x_n, \theta) = h(x_{n+1}; x_n, \theta), \tag{14}$$

where the functions f and h are non-degenerate probability densities (with respect to the first argument) dependent on the parameter and state vectors θ and x_n.

The joint density of the complete data (i.e. observable vector and latent state variables), due to the Markovian property in equations (13) and (14), can be written as,

$$p(\mathbf{z}_N, \mathbf{x}_N|\theta) = \left(\prod_{n=1}^{N} f(z_n; x_n, \theta)\right)\left(\prod_{n=2}^{N} h(x_n; x_{n-1}, \theta)\right) p(x_1|\theta). \tag{15}$$

Therefore, the observable data density is,

$$p(\mathbf{z}_N|\theta) = \int p(\mathbf{z}_N, \mathbf{x}_N|\theta) \mathrm{d}\mu(\mathbf{x}_N), \tag{16}$$

which falls under the general setting described in equation (9). The central idea of the Durbin-Koopman (DK) methodology consists in building a Gaussian linear State Space Model (SSM) for $\mathbf{z}_N$ in order to obtain a feasible (Gaussian) auxiliary sampler $g(\mathbf{x}_N|\mathbf{z}_N, \theta_g)$. Assuming a model was already built, the density of the observable data, equation (16), can be re-written as[9]

$$p(\mathbf{z}_N|\theta) = g(\mathbf{z}_N|\theta_g) \int \frac{p(\mathbf{z}_N, \mathbf{x}_N|\theta)}{g(\mathbf{z}_N, \mathbf{x}_N|\theta_g)} g(\mathbf{x}_N|\mathbf{z}_N, \theta_g) \, \mathrm{d}\mathbf{x}_N. \tag{17}$$

It is interesting to note that in this setting, the density of the observable data is expressed as the product of an (approximating) Gaussian density, that can easily be computed using the Kalman filter, by an adjusting factor, that can be estimated by simulation. The closer the joint importance density $g(\mathbf{z}_N, \mathbf{x}_N|\theta_g)$ is to the non-Gaussian density $p(\mathbf{z}_N, \mathbf{x}_N)$ the smaller will be the simulated sample required to attain a given accuracy level.

The parameters θ_g of the approximating linear Gaussian SSM are appropriately chosen to ensure that the mode of the implied posterior density $g(\mathbf{x}_N|\mathbf{z}_N, \theta_g)$ equals the 'true' posterior mode $\mathbf{x}^* = (x_1^{*\prime}, \ldots, x_N^{*\prime})'$ (i.e. the mode of the non-Gaussian posterior density $p(\mathbf{x}_N|\mathbf{z}_N, \theta)$). This can be achieved using the procedure described in Durbin and Koopman (2001, Chapter 11), which, for the MLFI model, is based on the linearization of the observational log-density using a second-order Taylor polynomial. The details of the application of this procedure to the class of MLFI models are reviewed later in this section.

[9]For simplicity of exposition I shall assume from now on that $\mathbf{x}_N$ is continuously distributed over $\mathbb{R}^{Nq}$ (as it is the case for the MLFI model), with q the dimension of the state vector x, and μ is the Lebesgue measure.

I now proceed by explicitly expressing the MLFI model, as described in section 2, in state space form. Consider the vector

$$z_n = (\tau_n \ , \ Y_1^1(t_n) \ , \ \dots \ , \ Y_S^K(t_n) \ , \ \iota^1(t_n)' \ , \ \dots \ , \ \iota^K(t_n)')'.$$

The vector z_n contains all observable information on the $\tilde{S}$ point processes at the event-point t_n ($\iota^k(t_n) = (\Delta N_1^k(t_n), \dots, \Delta N_S^k(t_n))'$ is unit k's random selection vector). The observational log-density is

$$\ln p(z_n|\psi_n, \mathcal{F}^o_{t_n^-}) = \sum_{s=1}^{S}\sum_{k=1}^{K} \Delta N_s^k(t_n) \ln\{\lambda_s^k(t_n)\} - Y_s^k(t_n) \int_{t_{n-1}}^{t_n} \lambda_s^k(t)\mathrm{d}t, \tag{18}$$

for $n = 1, \dots, N$ (set $N = \bar{N}(T) + 1$).

All fixed unknown coefficients entering the conditional hazard (2), jointly with the latent dynamic process ψ, can be placed in the state vector,

$$x_n = (\eta_1, \dots, \eta_S, \gamma_1', \dots, \gamma_S', \psi'(t_n))'. \tag{19}$$

By defining the deterministic (row vector) coefficients

$$Z_{sn}^k = (e_s', e_s' \otimes w^k(t_n)', \alpha_s),$$

the conditional hazard rate entering (18) can be written as $\lambda_s^k(t) = \exp\left(Z_{sn}^k x_n\right) \cdot H_s^k(t)$. In this context, $Z_{sn}^k x_n$ is the *signal* corresponding to the element $\Delta N_s^k(t_n)$ of $\iota^k(t_n)$. The state vector x_n can be modelled by the linear Markovian process

$$x_n = F_n x_{n-1} + R_n \varepsilon_n, \qquad \varepsilon_n \sim \mathrm{NIID}(0, Q_n), \qquad n = 1, 2, \dots, N(T), \tag{20}$$

with initial condition $x_0 \sim \mathrm{N}(a, P)$. The vector a and the matrix processes F_n, R_n, Q_n and P are predictable with respect to the observable filtration $\mathcal{F}_t^o$ and may depend on the parameter vector θ. If the vector x_n only consists of fixed unknown coefficients, we set $a = 0$, $F_n = R_n = I$, $Q_n = 0$ and $P = \kappa I$, where κ is the so-called diffuse prior constant. Usually, κ is set to some large value in numerical software, see Harvey (1989, pp. 367-8). Exact solutions for $\kappa \to \infty$ are available as well, see Durbin and Koopman (2001, Ch. 4). If the vector x_n only contains the latent autoregressive process (for example under equation 3), that is $x_n = \psi_n$, we set $a = 0$, $F_n = \mathrm{diag}(\rho_1^{\Delta N_1(t_n)}, \dots, \rho_S^{\Delta N_S(t_n)})$, $R_n = \iota(t_n)$, $Q_n = 1$ and $P = \mathrm{diag}((1-\rho_1^2)^{-1}, \dots, (1-\rho_S^2)^{-1})$. Here $\mathrm{diag}(d_1, \dots, d_S)$ denotes a square diagonal matrix of order S, containing the vector $(d_1, \dots, d_S)$ on the main diagonal. A combination of unknown coefficients and latent time series processes can be incorporated in (20) in a straightforward way. For example, in the case of (3) with $w^k(t) \equiv 0$, we have $x_n = (\eta_1 \ , \ \dots \ , \eta_S \ , \psi_n)'$ with $a = 0$, and

$$F_n = \begin{bmatrix} I_S & 0 \\ 0 & \mathrm{diag}(\rho_1^{\Delta N_1(t_n)}, \dots, \rho_S^{\Delta N_S(t_n)}) \end{bmatrix}, R_n = \begin{bmatrix} 0 \\ \iota(t_n) \end{bmatrix}, \quad Q_n = 1,$$

$$P = \begin{bmatrix} \kappa I_S & 0 \\ 0 & \text{diag}((1-\rho_1^2)^{-1}, \ldots, (1-\rho_S^2)^{-1}) \end{bmatrix}.$$

Equations (20) and (18) constitute a nonlinear, non-Gaussian state space model.

For obtaining the importance sampling density $g(\mathbf{x}_N|\mathbf{z}_N, \theta_g)$, the key idea is to build a linear Gaussian SSM for the set of rating event indicators at the event-point t_n,

$$\left\{\iota^1(t_n), \ldots, \iota^K(t_n)\right\}.$$

The observation equation of this approximating model takes the form,

$$\iota^k(t_n) = c_n^k + Z_n^k x_n + \xi_n^k, \qquad \xi_n^k \sim \text{NIID}(\mathbf{0}, diag(C_n^k)), \tag{21}$$

for $n = 1, \ldots, N$ and $k = 1, \ldots, K$ with the matrix $Z_n^k = (Z_{1n}^{k\prime}, \ldots, Z_{Sn}^{k\prime})'$. The s-dimensional vectors c_n^k and C_n^k are auxiliary constants to be determined (and constitute θ_g). Equations (20) and (21) make up a linear Gaussian SSM. The auxiliary vectors c_n^k and C_n^k are to be constructed in such a way that the mode of the posterior Gaussian approximating density $g(\mathbf{x}_N|\mathbf{z}_N)$ equals the true posterior mode $\mathbf{x}^*$. All these unknown vectors can be jointly determined using the recursive procedure described in Durbin and Koopman (2001, Chapter 11). For implementing this procedure an initial guess for the mode $\mathbf{x}^*$ has to be found. Denote this by $\mathbf{x}^{*(0)}$. The linear Gaussian SSM (21) is constructed for a given j by

$$\begin{aligned} c_n^k &= \iota^k(t_n) - Z_n^k\, x_n^{*(j)} - C_n^k\, Z_n^k\, \mathrm{k}_n, \\ C_n^k &= \left[Z_n^k\, \mathrm{K}_n\, Z_n^{k\prime}\right]^{-1}, \end{aligned} \tag{22}$$

where

$$\mathrm{k}_n = \left.\frac{\partial \ln p(\mathbf{z}_N|\mathbf{x}_N, \theta)}{\partial x_n}\right|_{\mathbf{x}_N = \mathbf{x}^{*(j)}},$$

$$\mathrm{K}_n = -\left.\frac{\partial^2 \ln p(\mathbf{z}_N|\mathbf{x}_N, \theta)}{\partial x_n \partial x_n'}\right|_{\mathbf{x}_N = \mathbf{x}^{*(j)}}.$$

A new guess of the mode $\mathbf{x}^{*(j+1)}$ is obtained by estimating the conditional mean of the state vector under the approximating linear Gaussian SSM (21) and (20). Because for the linear Gaussian SSM the conditional mean and mode of $\mathbf{x}_N$ coincide, this can be computed by the Kalman filter and smoothing algorithm (KFS). New guesses for the mode are obtained by the KFS based on (22) for $j = 1, 2, \ldots$ until convergence is reached according to some metric. Usually convergence takes place after 5 to 10 iterations. After convergence, the joint importance density $g(\mathbf{z}_N, \mathbf{x}_N, \theta_g^*)$ is obtained from (21) and (20) by evaluating (22) at the resulting (approximate value of the) posterior mode $\mathbf{x}^*$. Random draws $\mathbf{x}_N^m$ from $g\left(\mathbf{x}_N|\mathbf{z}_N, \theta_g^*\right)$, with $m = 1, \ldots, M$, can then be obtained using (for example) the simulation smoothing algorithm

of Durbin and Koopman (2002).

Because, for the MLFI model, the state equation (20) is Gaussian, the MC likelihood estimator (11) becomes

$$\hat{L}_{DK}\left(\theta|\mathbf{z}_N\right) = g(\mathbf{z}_N|\theta_g)M^{-1}\sum_{m=1}^{M}\frac{\prod_{n=1}^{N}p(z_n|\psi_n^m,\mathcal{F}^o_{t_n^-})}{g(\mathbf{z}_N|\mathbf{x}_N^m)}. \tag{23}$$

The parameter vector θ is obtained through the numerical optimization of (23), for example, using a quasi-Newton algorithm. To ensure a smooth surface in θ, the use of the so-called Common Random Numbers (CRN) approach is critical. This consists in using the same set of canonical random draws (in this case, realizations from the white Gaussian noise ε_n in (20)) to generate the samples $\{\mathbf{x}_N^m\}$ with $m = 1, \ldots, M$ used in the search for the optimal θ.

The state vector x_n contains fixed unknown coefficients and the dynamic latent process ψ. Estimating the posterior mean of the state vector leads to estimates of regression parameters and of the full path of the latent process $\Psi_{\bar{N}}(T)$. A straightforward estimate of the state vector, given the data, is obtained by weighting each random draw of the state vector $\mathbf{x}_N^m$ by its contribution to the likelihood function, that is

$$\hat{\mathbf{x}}_N = \left(\sum_{m=1}^{M} w_m\ \mathbf{x}_N^m\right) \Bigg/ \left(\sum_{m=1}^{M} w_m\right), \tag{24}$$

where

$$w_m = \left\{\prod_{n=1}^{N} p(z_n|\psi_n^m,\mathcal{F}^o_{t_n^-})\right\} \Bigg/ g(\mathbf{z}_N|\mathbf{x}_N^m). \tag{25}$$

Standard errors for $\hat{\mathbf{x}}_N$ are obtained by taking the square root of

$$\left[\left\{\sum_{m=1}^{M} w_m\ \mathbf{x}_N^m \odot \mathbf{x}_N^m\right\} \Bigg/ \left(\sum_{m=1}^{M} w_m\right)\right] - \hat{\mathbf{x}}_N \odot \hat{\mathbf{x}}_N. \tag{26}$$

3.2 The Efficient Importance Sampling algorithm

This section discusses the application of the EIS technique of Liesenfeld and Richard (2003) and Richard and Zhang (2007) to the estimation of MLFI models. The EIS algorithm is slightly more general than the DK methodology, in the sense that instead of assuming the model under study has the structure described in (13) and (14) it is only required that

$$p(z_n|x_n, \mathbf{x}_{n-1}, \mathbf{z}_{n-1}) = p(z_n|x_n, \mathbf{z}_{n-1}). \tag{27}$$

Therefore, the latent vector x no longer represents the full *state vector* of the system under study. Instead, under the current approach x simply represents a set of relevant unobserved processes. Additionally, (27) implies that the mechanism through which the latent processes

affect the observables is Markovian (i.e. memoryless), in the sense that dependence of observables on unobservables is strictly a contemporaneous phenomenon. Under these assumptions the observable data density can be written as

$$p(\mathbf{z}_N|\theta) = \int \left[\prod_{n=2}^{N} p(z_n|x_n, \mathbf{z}_{n-1}) p(x_n|\mathbf{x}_{n-1}, \mathbf{z}_{n-1})\right] p(z_1|x_1)p(x_1)\mathrm{d}\mathbf{x}_N. \tag{28}$$

The initial conditions are expressed by $p(z_1|x_1)p(x_1)$. The natural Monte Carlo estimator of the likelihood function $L(\theta|\mathbf{z}_N) = p(\mathbf{z}_N|\theta)$, as discussed in (10), in the current context is given by

$$\hat{L}_{NS}(\theta|\mathbf{z}_N) = M^{-1} \sum_{m=1}^{M} \left[p(z_1|x_1^m) \prod_{n=2}^{N} p(z_n|x_n^m, \mathbf{z}_{n-1})\right], \tag{29}$$

where, for every $m = 1, \ldots, M$, the sequence $\{x_n^m\}_{n=1}^N$ denotes a trajectory drawn from the natural samplers

$$p(x_1), p(x_2|x_1, z_1), \ldots, p(x_N|\mathbf{x}_{N-1}, \mathbf{z}_{N-1}). \tag{30}$$

However, as already mentioned, this estimator of the likelihood function is very inefficient. The latent variables are sampled directly from the statistical specification of the model without taking into account the relevant information contained in the observed values of z. As mentioned at the beginning of this section, one solution is to sample the latent vector $\mathbf{x}_N$ conditionally on the observations $\mathbf{z}_N$ using an auxiliary density. The EIS algorithm aims to construct a sequence of auxiliary samplers

$$s(x_1|\varphi_1), s(x_2|x_1;\varphi_2) \ldots, s(x_N|\mathbf{x}_{N-1};\varphi_N), \tag{31}$$

selected from a given parametric family of densities and indexed by the auxiliary parameters $\varphi_1, \ldots, \varphi_N$.

Recalling the discussion at the beginning of section 3, it becomes clear that if it would be possible to obtain a sequence of auxiliary samplers (31) such that its product $s(x_1|\varphi_1)\prod_{n=2}^{N} s(x_n|x_{n-1};\varphi_n)$ is proportional (as a function of $\mathbf{x}_N$) to the product of the factors in (28) involving $\mathbf{z}_N$, then the unbiased MC estimator (11) would be equal a.s. to the likelihood of the observable data i.e. the MC variance of this estimator would be zero. Therefore, the tuning parameters φ are to be chosen in a such a way as to minimize (inside the chosen family of s densities, that is) the MC variance of the importance sampling MC estimator of the likelihood (11). In the current context, this estimator can be written as,

$$\hat{L}_{IS}(\theta|\mathbf{z}_N) = M^{-1} \sum_{m=1}^{M} \left[\frac{p(z_1|x_1^m)p(x_1^m)}{s(x_1^m|\varphi_1)} \prod_{n=2}^{N} \frac{p(z_n|x_n^m, \mathbf{z}_{n-1})p(x_n^m|\mathbf{x}_{n-1}^m, \mathbf{z}_{n-1})}{s(x_n^m|\mathbf{x}_{n-1}^m;\varphi_n)}\right], \tag{32}$$

where $\mathbf{x}_N^m$ denotes a full trajectory drawn from the importance samplers (31). Minimizing the MC variance of (32) can be achieved by trying to 'match' the product in the numerator

with that in the denominator. That is, minimizing the difference between the numerator and the product of the denominator by a constant. This minimization problem (which has to be solved for each θ) takes place in a very high-dimensional space. The key idea behind the EIS algorithm consists in breaking this problem down into a set of low-dimensional minimization subproblems. The structure of the likelihood, as seen in the 'one-step ahead' factorization (28), suggests decomposing the large minimization problem into a sequence of subproblems, one for each time period. Note, however, that for any given values of $\mathbf{z}_N$ and θ, the integral of

$$p(z_n|x_n, \mathbf{z}_{n-1})p(x_n|\mathbf{x}_{n-1}, \mathbf{z}_{n-1})$$

with respect to x_n does depend upon $\mathbf{x}_{n-1}$, while that of $s(x_n|\mathbf{x}_{n-1}; \varphi_n)$ equals 1 by definition. This shows that it is not possible to secure a 'good match' between the factors in the numerator and in the denominator period by period, independently from one another.

This problem was solved in Liesenfeld and Richard (2003) and Richard and Zhang (2007) by explicitly writing the importance samplers $s(x_n|\mathbf{x}_{n-1}; \varphi_n)$ as the ratio of a density kernel $\kappa(\mathbf{x}_n; \varphi_n)$ and the corresponding integrating constant $\chi(\mathbf{x}_{n-1}; \varphi_n)$, that is,

$$s(x_n|\mathbf{x}_{n-1}; \varphi_n) = \frac{\kappa(\mathbf{x}_n; \varphi_n)}{\chi(\mathbf{x}_{n-1}; \varphi_n)},$$

with $\chi(\mathbf{x}_{n-1}; \varphi_n) = \int \kappa(\mathbf{x}_n; \varphi_n)\mathrm{d}x_n$. Noticing that making $\kappa(\mathbf{x}_n; \varphi_n)$ as close as possible to being proportional to $p(z_n|x_n, \mathbf{z}_{n-1})p(x_n|\mathbf{x}_{n-1}, \mathbf{z}_{n-1})$ would leave $\chi(\mathbf{x}_{n-1}; \varphi_n)$ unaccounted for, suggests sending it back to moment's $n-1$ minimization problem. The presence of the $(n+1)^{th}$ integrating constant at the nth optimization problem binds together the entire sequence.

In practice, the auxiliary samplers are obtained by recursively solving a sequence of low-dimensional least squares problems (with $n = N, \ldots, 1$) of the form,

$$\hat{\varphi}_n(\theta) = arg \min_{\varphi_n} \sum_{m=1}^{M} \left\{ \ln \left[\frac{p(z_n|x_n^m, \mathbf{z}_{n-1})p(x_n^m|\mathbf{x}_{n-1}^m, \mathbf{z}_{n-1})\chi(\mathbf{x}; \hat{\varphi}_{n+1})}{c_n \kappa(\mathbf{x}_n^m; \varphi_n)} \right] \right\}^2, \tag{33}$$

with the auxiliary condition $\chi(\mathbf{x}_N^m; \hat{\varphi}_{N+1}) = 1$.

Summarizing, the EIS algorithm works through the following 3 steps:

1. Generate M trajectories $\mathbf{x}_N^m$, $m = 1, \ldots, M$, drawn from the natural samplers (30).

2. For each n (from N to 1) estimate the parameters φ_n of the auxiliary regression (33) by least-squares (over the sample made up of the M observations obtained in the previous step).

3. Generate M trajectories $\mathbf{x}_N^m$, $m = 1, \ldots, M$, from the auxiliary samplers (31) corresponding to the sequence of auxiliary parameters φ_n with $n = 1, \ldots, N$.

The last 2 steps should be iterated until the estimates of the auxiliary parameters φ_n converge, according to some metric. After convergence, the last set of trajectories $\mathbf{x}_N^m$ are used to estimate the likelihood with the EIS estimator (32). Liesenfeld and Richard (2003) state that convergence typically takes around 5 iterations.

Application of the EIS algorithm to the MLFI class of models is feasible as the one-step ahead observational density (18) falls under the scope of condition (27). The latent vector $\mathbf{x}_N$, with $N = \bar{N}(T)+1$, can be equated to the full path of the latent process $\Psi_{\bar{N}(T)}$. While the one-step-ahead latent density $p(x_n|\mathbf{x}_{n-1}, \mathbf{z}_{n-1})$ is independent of both $\mathbf{z}_{n-1}$ and $\mathbf{x}_{n-2}$, and is given by either (3) or (4). As for the Durbin-Koopman method, all draws from the auxiliary samplers should be obtained using the technique of CRN.

The final element required to apply the EIS algorithm to the estimation of the MLFI model is the choice of the auxiliary samplers (31). The density of the latent vector $\mathbf{x}_N$ in the MLFI model is Gaussian, this suggests using Gaussian densities as the importance densities (31). Because the class of Gaussian densities is closed under multiplication, and our objective is 'matching' the denominator with the numerator in (32), the natural samplers are to be included in $s(x_n|x_{n-1}; \varphi_n)$. In fact, the auxiliary samplers for the MLFI model are completely identical to those used in Liesenfeld and Richard (2003) for the Stochastic Volatility model, and in Bauwens and Hautsch (2006) for the (single-state) Stochastic Conditional Intensity model. The mean $\tilde{\mu}_n$ and variance $\tilde{\sigma}_n^2$ of these Gaussian auxiliary samplers at time-step n are given by

$$\begin{aligned} \tilde{\sigma}_n^2 &= (1/\sigma_n^2 - 2\varphi_{2,n})^{-1} \\ \tilde{\mu}_n &= (\frac{\mu_n}{\sigma_n^2} + \varphi_{1,n})\tilde{\sigma}_n^2, \end{aligned} \tag{34}$$

where μ_n and σ_n^2 represent the conditional mean and variance, respectively, of the latent process x over the spell $(t_n, t_{n+1}]$. The auxiliary parameters $\varphi_{1,n}$ and $\varphi_{2,n}$ are obtained from the least squares problems (33). Under the current choice for the importance samplers, these minimization problems are linear over the auxiliary parameters φ and are, therefore, solvable by the ordinary least squares method (subject to a positivity constraint over $\tilde{\sigma}_n^2$).

Computationally, the first step of the EIS algorithm in the very first iteration is still inefficient. Furthermore, when the natural samplers correspond to an integrated process of at least order one, drawing long trajectories from such a process may lead to numerical instability.

In fact, instead of drawing the initial set of trajectories from the natural samplers, we can use a sequence of samplers that already takes into account the observed data. Expressing the observational density (18) as a second order Taylor polynomial in $\psi(t) = (\psi(t) - \mathrm{E}[\psi(t)])$, provides the moments of a sequence of auxiliary Gaussian samplers that can be used as the initial samplers.

Besides obtaining estimates of the parameter vector θ, by maximizing the MC estimator of the likelihood function (32), it is also important to be able to obtain smoothed and filtered estimates of the latent vector $\mathbf{x}_N$. In general terms, we are interested in computing filtered estimates of an arbitrary function ϕ of the latent process x,

$$\mathrm{E}\left[\phi\left(x_n\right)|\mathbf{z}_{n-1}\right] = \frac{\int \phi\left(x_n\right) p\left(x_n|\mathbf{z}_{n-1},\mathbf{x}_{n-1},\theta\right) p\left(\mathbf{z}_{n-1},\mathbf{x}_{n-1}|\theta\right) \mathrm{d}\mathbf{x}_n}{\int L\left(\theta|\mathbf{z}_{n-1},\mathbf{x}_{n-1}\right) \mathrm{d}\mathbf{x}_{n-1}}, \tag{35}$$

with $n = 1, \ldots, N$. The dependence in θ is removed by inserting in its place the corresponding SML estimate. The denominator in the ratio of integrals above is simply the Likelihood associated with the first $n-1$ components of $\mathbf{z}_N$. Accordingly it can be estimated using the MC estimator (32) with the obvious modification.

The numerator can be estimated with

$$M^{-1} \sum_{m=1}^{M} \left\{ \phi\left(x_n^m\right) \prod_{i=1}^{n-1} \frac{p(z_i|x_i^m, \mathbf{z}_{(i-1)}) p(x_i^m|\mathbf{x}_{(i-1)}^m, \mathbf{z}_{(i-1)})}{s(x_i^m|\mathbf{x}_{(i-1)}^m;\varphi_i)} \right\} \tag{36}$$

where $\mathbf{x}_{n-2}^m$ denotes a trajectory drawn from the sequence of auxiliary samplers associated with $L\left(\theta|Z_{n-1}\right)$ and x_n^m is a draw from

$$p(x_n|\mathbf{x}_{n-1}^m, \mathbf{z}_{n-1}).$$

This procedure is computationally demanding, as it requires tuning a new sequence of auxiliary samplers for each n.

We can of course also obtain smoothed estimates of any functional Φ of $\mathbf{x}$. The corresponding procedure being less demanding in computational terms than the one outlined above for the filtered estimates,

$$\mathrm{E}\left[\Phi(\mathbf{x}_N)|\mathbf{z}_N\right] = \frac{\int \Phi\left(\mathbf{x}_N\right) \prod_{n=1}^{N} p(z_n|x_n, \mathbf{z}_{n-1}) p(x_n|\mathbf{x}_{n-1}, \mathbf{z}_{(i-1)}) \mathrm{d}\mathbf{x}_N}{L\left(\theta \mid \mathbf{z}_N\right)}.$$

The denominator is simply the unconditional data likelihood associated with θ. As before, this dependence is eliminated by inserting in its place the ML estimate. We can estimate the integral in the numerator by using the same set of random draws from the auxiliary samplers (obtained at the end of the EIS iterations) which were already used to estimate the unconditional likelihood at the ML estimate of θ. This is done using

$$M^{-1} \sum_{m=1}^{M} \left[\Phi\left(\mathbf{x}_N^m\right) \prod_{n=1}^{N} \frac{p(z_n|x_n^m, \mathbf{z}_{n-1}) p(x_n^m|\mathbf{x}_{n-1}^m, \mathbf{z}_{n-1})}{s(x_n^m|\mathbf{x}_{n-1}^m;\varphi_n)} \right], \tag{37}$$

where $\mathbf{x}_N^m$ denotes one full trajectory drawn from the auxiliary samplers (31) associated with $\hat{\theta}_{ML}$. In this discussion, and for simplicity, the initial conditions were left implicit. That is I assumed

$$p(z_1|x_1^m, \mathbf{z}_{(0)}) p(x_1^m|\mathbf{x}_{(0)}^m, \mathbf{z}_{(0)}) \equiv p(z_1|x_1) p(x_1), \qquad s(x_1^m|\mathbf{x}_{(0)}^m;\varphi_1) \equiv s(x_1|\varphi_1).$$

4 Approximate Maximum Likelihood estimation

This section reviews the approximate Maximum Likelihood approach introduced in Davis and Rodriguez-Yam (2005) and adapts it to the estimation of MLFI models. The main idea of this method is to construct a closed-form approximation to the integral in the true data likelihood (9) by approximating the structural density (8) using a second order Taylor expansion centred on the mode of the posterior density (12). The main advantage of this method is its (relative) computational simplicity and resulting speed. No use is made of Monte Carlo integration techniques. The closed-form approximation to the data density can be directly maximized (numerically) to yield approximate ML estimates.

Being only an approximate method implies that on top of the (unavoidable) sampling error there is an additional error term in the resulting estimators. However, the same holds with all MC based methods, in particular the ones covered in this paper. The only difference resides in the fact that while the later resort to (pseudo) random number generators to construct their (pseudo random) approximation to the likelihood function, the present method relies completely on deterministic numerical methods for achieving the same objective. Theoretically, the effectiveness of this method depends on whether or not the posterior density implied by the model is approximately Gaussian.

The second limitation of the method is that it can only be applied to the special case where the state transition density (14) is Normal. However this still covers a very large class of non-Gaussian (observation) state space models. The MLFI is one such case.

Consider the model described in equations (13) and (14). Assume that the state equation (14) takes the linear Gaussian form presented in (20). Under this setting the joint density of the 'complete data,' equation (15), becomes

$$p(\mathbf{z}_N, \mathbf{x}_N|\theta) = p(\mathbf{z}_N|\mathbf{x}_N;\theta)\frac{|\mathbf{V}|^{1/2}}{(2\pi)^{N/2}}\exp\left[-(\mathbf{x}_N-\mu)'\mathbf{V}(\mathbf{x}_N-\mu)/2\right], \tag{38}$$

where $\mathbf{V}^{-1} = \operatorname{cov}\{\mathbf{x}_N\}$ and $\mu = \mathrm{E}[\mathbf{x}_N]$. The covariance $\operatorname{cov}\{\mathbf{x}_N\}$ and the mean of the state vector (20) can be computed analytically.

Let $l(\theta;\mathbf{z}_N|\mathbf{x}_N) = \ln p(\mathbf{z}_N|\mathbf{x}_N;\theta)$. The loglikelihood associated with the complete data resulting from (38) is

$$l(\theta;\mathbf{z}_N,\mathbf{x}_N) = l(\theta;\mathbf{z}_N|\mathbf{x}_N) + \frac{1}{2}\ln|\mathbf{V}| - \frac{N}{2}\ln(2\pi) - \frac{1}{2}(\mathbf{x}_N-\mu)'\mathbf{V}(\mathbf{x}_N-\mu). \tag{39}$$

Denote the gradient[10] of $l(\theta;\mathbf{z}_N|\mathbf{x}_N)$ evaluated at the so-called posterior mode of the state vector, i.e. the mode $\mathbf{x}^* = \mathbf{x}^*(\mathbf{z}_N;\theta)$ of $p(\mathbf{x}_N|\mathbf{z}_N;\theta)$, by

$$\mathbf{k}^* \equiv \frac{\partial}{\partial \mathbf{x}_N} l(\theta;\mathbf{z}_N|\mathbf{x}_N)|_{\mathbf{x}_N=\mathbf{x}^*}.$$

[10]Here I represent the gradient as a column and not as a row-vector as it is usual.

Note that $\mathbf{x}^*$ solves the nonlinear (vector) equation

$$\frac{\partial}{\partial \mathbf{x}_N} l(\theta; \mathbf{z}_N, \mathbf{x}_N) = \mathbf{0}, \tag{40}$$

leading to $\mathbf{k}^* = \mathbf{V}(\mathbf{x}^* - \mu)$. Let $R(\mathbf{x}_N; \mathbf{x}^*)$ denote the error associated with the second order Taylor expansion of $l(\theta; \mathbf{z}_N|\mathbf{x}_N)$ centred on $\mathbf{x}^*$,

$$\begin{aligned} l(\theta; \mathbf{z}_N|\mathbf{x}_N) &= \mathbf{l}^* + (\mathbf{x}_N - \mathbf{x}^*)'\mathbf{k}^* - \frac{1}{2}(\mathbf{x}_N - \mathbf{x}^*)'\mathbf{K}^*(\mathbf{x}_N - \mathbf{x}^*) \\ &\quad + R(\mathbf{x}_N; \mathbf{x}^*), \end{aligned} \tag{41}$$

where $\mathbf{l}^* = l(\theta; \mathbf{z}_N|\mathbf{x}_N)|_{\mathbf{x}_N = \mathbf{x}^*}$ and

$$\mathbf{K}^* \equiv -\frac{\partial}{\partial \mathbf{x}_N \partial (\mathbf{x}_N)'} l(\theta; \mathbf{z}_N|\mathbf{x}_N)|_{\mathbf{x}_N = \mathbf{x}^*}.$$

Substituting (41) in (39) and grouping all the terms containing $\mathbf{x}_N$ into a single quadratic form yields,

$$\begin{aligned} l(\theta; \mathbf{z}_N, \mathbf{x}_N) &= -\frac{N}{2}\ln(2\pi) + \frac{1}{2}\ln|\mathbf{V}| + \mathbf{l}^* - \frac{1}{2}(\mathbf{x}^* - \mu)'\mathbf{V}(\mathbf{x}^* - \mu) \\ &\quad -\frac{1}{2}(\mathbf{x}_N - \mathbf{x}^*)'(\mathbf{K}^* + \mathbf{V})(\mathbf{x}_N - \mathbf{x}^*) + R(\mathbf{x}_N; \mathbf{x}^*). \end{aligned} \tag{42}$$

Recalling that

$$p(\mathbf{x}_N|\mathbf{z}_N; \theta) = \frac{p(\mathbf{z}_N, \mathbf{x}_N|\theta)}{\int p(\mathbf{z}_N, \mathbf{x}_N|\theta)\mathrm{d}\mathbf{x}_N},$$

let $p_a(\mathbf{x}_N|\mathbf{z}_N; \theta)$ denote the (approximate) posterior density based on the loglikelihood given in (42) when the error term $R(\mathbf{x}_N; \mathbf{x}^*)$ is dropped. It follows that p_a is a Gaussian density with mean $\mathbf{x}^*$ and covariance matrix $(\mathbf{K}^* + \mathbf{V})^{-1}$.

The likelihood associated with the observable data, equation (16), can be written as

$$\begin{aligned} L(\theta; \mathbf{z}_N) &= \frac{|\mathbf{V}|^{\frac{1}{2}}}{|\mathbf{K}^* + \mathbf{V}|^{\frac{1}{2}}} e^{\mathbf{l}^* - \frac{1}{2}(\mathbf{x}^* - \mu)'\mathbf{V}(\mathbf{x}^* - \mu)} \int p_a(\mathbf{x}_N|\mathbf{z}_N; \theta) e^{R(\mathbf{x}_N; \mathbf{x}^*)} \mathrm{d}\mathbf{x}_N. \\ &= L_a(\theta; \mathbf{z}_N)\mathrm{Err}_a(\theta; \mathbf{z}_N), \end{aligned} \tag{43}$$

where $L_a(\theta; \mathbf{z}_N)$ is the approximation to the (true) likelihood function $L(\theta; \mathbf{z}_N)$,

$$L_a(\theta; \mathbf{z}_N) = \frac{|\mathbf{V}|^{\frac{1}{2}}}{|\mathbf{K}^* + \mathbf{V}|^{\frac{1}{2}}} \exp\left[\mathbf{l}^* - \frac{1}{2}(\mathbf{x}^* - \mu)'\mathbf{V}(\mathbf{x}^* - \mu)\right], \tag{44}$$

that is obtained when the factor $\exp[R(\mathbf{x}_N; \mathbf{x}^*)]$ is dropped in the integral in (43).

The *approximation error* implied is

$$\mathrm{Err}_a(\theta; \mathbf{z}_N) = \int p_a(\mathbf{x}_N|\mathbf{z}_N; \theta) e^{R(\mathbf{x}_N; \mathbf{x}^*)} \mathrm{d}\mathbf{x}_N = \mathrm{E}_{p_a}\left[e^{R(\mathbf{x}_N; \mathbf{x}^*)}|\mathbf{z}_N\right]. \tag{45}$$

If $p_a(\mathbf{x}_N|\mathbf{z}_N; \theta)$ is highly concentrated around $\mathbf{x}^*$, then this multiplier is close to one.

Numerical maximization of (44) with respect to θ yields the (approximate) ML estimate of the

parameter. An (informal) assessment of the quality of this approximation is given by inspecting the 'size' (expressed in terms of a matrix norm) of the matrix $(\mathbf{K}^* + \mathbf{V})^{-1}$ at the optimum.
In order to make the closed-form approximation (44) to the true likelihood function (43) operational, it is necessary to know (at least approximately) the value of the posterior mode $\mathbf{x}^*$. Applying the Newton-Raphson algorithm to the nonlinear equation (40) leads to the following recursive scheme

$$\mathbf{x}^{(i+1)} = (\mathbf{K}^{(i)} + \mathbf{V})^{-1} \left(\mathbf{k}^{(i)} + \mathbf{K}^{(i)}\mathbf{x}^{(i)} + \mathbf{V}\mu\right), \tag{46}$$

where

$$\mathbf{k}^{(i)} \equiv \frac{\partial}{\partial \mathbf{x}_N} l(\theta; \mathbf{z}_N|\mathbf{x}_N)|_{\mathbf{x}_N=\mathbf{x}^{(i)}} \text{ and } \mathbf{K}^{(i)} = -\frac{\partial}{\partial \mathbf{x}_N \partial (\mathbf{x}_N)'} l(\theta; \mathbf{z}_N|\mathbf{x}_N)|_{\mathbf{x}_N=\mathbf{x}^{(i)}}. \tag{47}$$

If the number of observations N is small, it is possible to use directly both the recursion (46) and the approximation (44). However if N is large, inverting the symmetric matrix $\mathbf{\Gamma} = \mathbf{K}^{(i)} + \mathbf{V}$ appearing in (46) and computing the determinant $|\mathbf{K}^* + \mathbf{V}|$ in (44) directly would be awkward. The solution to this problem consists in employing recursive computational procedures for the inversion of symmetrical positive definite matrices. One such procedure can be derived from the *innovations algorithm*[11]. The innovations algorithm provides the following particular case of the LU decomposition of $\mathbf{\Gamma}$,

$$\mathbf{\Gamma} = LDL', \tag{48}$$

where $D = diag(\nu_0, \ldots, \nu_{N-1})$ and L is the sum of the identity with a strictly lower triangular matrix. The innovations algorithm enables the recursive computation of the components of both D and L.
Setting $W^{(i)} = \left(\mathbf{k}^{(i)} + \mathbf{K}^{(i)}\mathbf{x}^{(i)} + \mathbf{V}\mu\right)$, we can write (46) as

$$\mathbf{x}^{(i+1)} = \mathbf{\Gamma}^{-1} W^{(i)} = (L')^{-1}\zeta^{(i)},$$

where the *normalized residuals* $\zeta^{(i)} = D^{-1}L^{-1}W^{(i)}$ are directly obtained from the output of the innovations algorithm. Because L is a lower triangular matrix, $\mathbf{x}^{(i+1)}$ is obtainable by solving the linear system $L'\mathbf{x}^{(i+1)} = \zeta^{(i)}$ by backward substitution. The determinant of $\mathbf{\Gamma}$ equals $|D| = \prod_{j=0}^{N-1} \nu_j$. Therefore, it is possible to implement the Newton-Raphson scheme for obtaining the posterior mode $\mathbf{x}^*$ without the need to invert directly a square matrix of order $N \times S \times (p+2)$ at each and every iteration.
In the particular case where the observation density belongs to the Exponential family, the approximation to the posterior density $p_a(\mathbf{x}_N|\mathbf{z}_N;\theta)$ coincides with the importance sampler

[11] For any given time series $\{X_n\}$ with finite second moments and zero mean, the innovations algorithm provides a recursive method for computing $\hat{X}_{n|n-1} = E[X_n|X_1, \ldots, X_{n-1}]$ from knowledge of the autocovariance function of $\{X_n\}$, see Proposition 5.2.2. in Brockwell and Davis (1991).

obtained using the DK methodology.

As before, our interest is not limited to the estimation of the parameter vector θ but also to the estimation of the full path $\mathbf{x}_N$ of the latent process given all the observations $\mathbf{z}_N$. The mean of the posterior density $p(\mathbf{x}_N|\mathbf{z}_N;\theta)$ is normally termed the *smoothed state vector*,

$$\mathrm{E}\left[\mathbf{x}_N|\mathbf{z}_N;\theta\right] = \frac{1}{L(\theta;\mathbf{z}_N)}\int \mathbf{x}_N p(\mathbf{z}_N,\mathbf{x}_N|\theta)\mathrm{d}\mathbf{x}_N. \tag{49}$$

Unlike the case of the parameter vector, estimation of the smoothed state vector requires the use of Monte Carlo integration with importance sampling. However, the procedure is relatively simple, because the importance sampling density $p_a(\mathbf{x}_N|\mathbf{z}_N;\theta)$ was already obtained during the estimation of the θ vector. The Monte Carlo estimator of the smoothed state vector is,

$$\frac{M^{-1}}{\hat{L}(\theta;\mathbf{z}_N)}\sum_{m=1}^{M}\mathbf{x}_N^m\frac{p(\mathbf{z}_N,\mathbf{x}_N^m|\theta)}{p_a(\mathbf{x}_N^m|\mathbf{z}_N;\theta)}. \tag{50}$$

Where the (observable) data density estimator in the denominator is,

$$\hat{L}(\theta;\mathbf{z}_N) = M^{-1}\sum_{m=1}^{M}\frac{p(\mathbf{z}_N,\mathbf{x}_N^m|\theta)}{p_a(\mathbf{x}_N^m|\mathbf{z}_N;\theta)}. \tag{51}$$

The common random samples $\mathbf{x}_N^m$, with $m = 1,\ldots,M$, in both estimators (51) and (50), are drawn from $p_a(\mathbf{x}_N|\mathbf{z}_N;\theta)$. The dependence on the unknown parameter vector θ is removed by inserting in its place the (approximate) ML estimate obtained from maximizing (44). Because in a Gaussian model the mean and the mode coincide, a second method for assessing the quality of the approximation $p_a(\mathbf{x}_N|\mathbf{z}_N;\theta)$ to the exact posterior density consists in inspecting the difference between the posterior mode $\mathbf{x}^*$ and the posterior mean of the state vector (50).

5 Simulation setup and results

In order to illustrate the three different estimation procedures discussed in this paper, I conducted a Monte Carlo study. This section discusses the set up of this simulation study and reports the corresponding results. Three different MLFI models were considered, each with four non-absorbing states (in the context of the empirical application in Koopman et al. 2006, 2008, these are interpretable as four different credit rating classes) and an absorbing one (modelling the default state). In all three cases there is a single unobserved intensity component ψ, no covariates and no duration dependence (i.e. $H_s^k(s) \equiv 1$). That is, for each statistical unit in the sample, the intensity specification (2) is restricted to

$$\tilde{\lambda}_s^k(t) = Y_s^k(t)\exp\left[\eta_s + \alpha_s\psi_s(t)\right] \quad s = 1,\ldots,16, \tag{52}$$

for the counting processes associated with all possible transitions from a non-absorbing state to another state (absorbing or not). The intensity processes are set to $\tilde{\lambda}_s^k(t) = 0$ for $s = 17, \ldots, 20$, for transitions from the absorbing state to the four non-absorbing ones.

I consider three different univariate specifications for the latent process. In the first two cases, the latent process ψ is specified as a standard univariate AR(1). The first AR(1) specification (model A) has a fixed autoregressive coefficient $\rho = 0.9$ corresponding to a case where the latent dynamics of the rating process are fairly persistent. In the second case (model B), an autoregressive coefficient of 0.5 is used, corresponding to a situation where the latent dynamics are less clear. In both these cases the latent disturbances are i.i.d. standard Normal. Finally, in the third model, model C, the latent process follows an autoregressive process with both a time-varying autocorrelation coefficient and a time-varying innovations variance, by using

$$\psi_{n+1} = \rho^{\tau_n}\psi_n + \sqrt{\frac{1-\rho^{2\tau_n}}{(1-\rho^2)}}\varepsilon_{n+1}, \qquad \varepsilon_{n+1} \sim \text{NIID}(0,1), \tag{53}$$

where I set $\rho = 0.9$. This brings the behaviour of the latent 'credit cycle' ψ close to that of an Ornstein-Uhlenbeck process sampled at the (discrete) set of 'event moments' $\{t_n\}$.

Unit (debt issuer) heterogeneity enters the three different model specifications through the different η_s and α_s parameters used for the different transition types $s = 1, \ldots, 16$. The parameter values chosen for the simulations appear in the first column of Tables 1, 2 and 3. Following the empirical application (and corresponding results) in Koopman et al. (2006, 2008), the constant baseline hazard parameters η were chosen in order to yield the typical structure seen in credit rating transition matrices. In this setting, the transitions corresponding to both large downgrades and upgrades typically have low intensity. This is expressed in lower values of η than those corresponding to those transitions between pairs of adjacent states (ratings). With regard to the sensitivity parameters α, on the contrary, I chose values that are fairly lower (in absolute terms) than the values reported in table 4 of Koopman et al. (2008). For a fixed (non-constant) path of the latent process ψ, the degree of clustering in observed transitions is directly proportional to the size (in absolute value) of the α parameters. Clearly, a faint signal is harder to recover than a strong one. Therefore, I chose the α's in order to yield a challenging signal extraction exercise.

ML estimation of the MLFI class of models is particularly demanding computationally, not only because the data density has no (exact) closed form, but also due to the large number of unknown parameters. These last ones have (at least the sensitivity α_s parameters and the autoregressive ρ_s coefficients, see discussion bellow) to be obtained through the numerical optimization of the loglikelihood function, regardless of whether this last one is estimated using MC methods or approximated using deterministic numerical methods. In any realistic real-

world application of the model, it is essential not only to allow one distinct baseline intensity η_s parameter for each type of transition but there should also be a reasonable number of independent sensitivity α_s parameters. In order to make the simulation set up reasonably realistic, I allow a distinct α_s parameter for each different transition type $s = 1, \ldots, 16$. This means that for all three models there is a total of 33 parameters to be estimated.

One difference between the three methods reviewed in this paper results from the need to use numerical optimization algorithms (for example quasi-Newton methods). Under the DK methodology discussed in Koopman et al. (2008) and reviewed in subsection 3.1, the size of the parameter vector is reduced to just 17, because the η_s parameters are loaded into the state vector x. Due to the similarities between the MLFI and the basic Stochastic Volatility (SV) models, I implement directly both the EIS algorithm, as discussed in Liesenfeld and Richard (2003), and the (deterministic) numerical procedure of Davis and Rodriguez-Yam (2005). This leads to a parameter vector of size 33. Nevertheless, the EIS algorithm can be slightly modified to treat fixed effects (like the η_s parameters) as part of the latent vector x (Richard and Zhang, 2007).

The data generating process (dgp) in all three models makes some of the transitions extremely unlikely (for example from rating class '1' to 'Default'). This means that in some of the simulated datasets there are no transitions of this type. In such cases both the η_s and α_s parameters are not identified and, therefore, cannot be estimated. Additionally, when there is a single transition of a given type, the corresponding α_s parameter is not identified as well. This means that not all η_s and α_s parameters can be estimated for every replication. Over a given simulated dataset, I chose to estimate both parameters only for those transition types for which at least three events were recorded. Tables 1, 2 and 3 report only the parameters for which a reasonable number of replications exist (at least 20 out of a total of 100 replications performed). This number is indicated in superscript and between parenthesis in the columns containing the MC averages.

For all three model specifications I consider $K = 100$ units (i.e. debt issuers) being observed over 30 periods of time. A panel of units and state (rating) transitions is generated as follows. At time $t_0 = 0$, the sample contains an equal number of units in each state (rating category). The unobserved process $\psi(t)$ is initialized at zero. Given the parameters, this completely specifies the intensities up to the event date t_1. For the time interval $(t_{n-1}, t_n]$, the intensity of the *pooled* process is defined by

$$\lambda^*(t_n) = \sum_{k=1}^{K} \sum_{s=1}^{S} \lambda_s^k(t_n), \tag{54}$$

with $\lambda^*(t_1)$ applicable over the first spell $(t_0, t_1]$. The length of any spell in the pooled process can therefore be drawn from the exponential distribution with intensity parameter $\lambda^*(t_n)$. Given the durations of the spells $(t_{n-1}, t_n]$ for $i = 1, \ldots, \bar{N}(T)$, the firm experiencing a rating event is drawn from the univariate Multinomial$\{\pi_1(t_n), \ldots, \pi_K(t_n)\}$ distribution where the probability of drawing unit k is given by

$$\pi_k(t_n) = [\lambda^*(t_n)]^{-1} \sum_{s=1}^{S} \lambda_s^k(t_n), \qquad k = 1, \ldots, K. \tag{55}$$

Next, the type of rating event for unit k is drawn from the multinomial distribution with the probability of state s being drawn for unit k given by

$$\pi_s^k(t_n) = \left[\sum_{s=1}^{S} \lambda_s^k(t_n) \right]^{-1} \lambda_s^k(t_n), \tag{56}$$

for $s = 1, \ldots, S$ and $k = 1, \ldots, K$. If the event is a default, the dummy variable $Y_s^k(t)$ jumps to zero. Finally, the unobserved common risk factor $\psi_i = \psi(t_i)$ is updated using either (3) (with $\rho = 0.9$, in the case of model A, or $\rho = 0.5$ for model B), or (4). In the later case I set $\rho = 0.9$. The disturbances ε_i, $i = 1, \ldots, \bar{N}(T)$, are drawn from a standard normal distribution in all three cases. This process is repeated until all units have entered the absorbing default state, or until the event time t_n exceeds the maximum number of time-periods (30).

The combination of parameters in the dgp with the number of units and the size of the time-window leads, typically, to a number of events per simulation between 500 and 600.

A rather imperfect[12] assessment of the computational efficiency of each one of the three methods is provided by the average amount of time required to estimate the unknown parameters in any given replication (using the quasi-Newton BFGS algorithm). These were, for model A, 15 minutes for the Durbin and Koopman (1997, 2000) SML methodology, 50 minutes for the approximate ML algorithm of Davis and Rodriguez-Yam (2005), and finally 90 minutes required in average by the EIS algorithm of Liesenfeld and Richard (2003) and Richard and Zhang (2007).[13]

[12]It must be stressed that while both the EIS and DY algorithms were directly implemented in a high-level language (`Ox` version 3.30), the `Ox` implementation of the DK methodology makes extensive use of the numerical routines contained in the package `SsfPack` (implemented in the lower level C language). Therefore, the comparison results reported in this paper, with regard to computational times when including the DK methodology, cannot be considered as a benchmark. The simulations were conducted on a standard desktop computer with 1 Gb of RAM and based around an Intel Pentium IV processor with a clock speed of 3.4 GHz running under Windows XP Professional version 2002 (Service Pack 2).

[13]In the `Ox` implementation of the DY-AML algorithm, a relative change threshold of 10^{-4} was used to assess convergence of the Newton-Raphson algorithm. To insure convergence of the auxiliary regression coefficients, 5 EIS iterations were used, starting from the 2^{nd} order Taylor polynomial samplers. Convergence of the numerical approximation to the posterior mode in the DK-SML algorithm was gauged using a tolerance of 10^{-7}.

All computations were made with the `Ox` matrix programming language of Doornik (2002) version 3.30. The DK methodology was implemented using the routines contained in the package `SsfPack` version beta 3.05.
I start by discussing the results obtained for model A. These are presented in Table 1.

<INSERT TABLE 1 ABOUT HERE>

For the majority of the baseline hazard parameters η, the most accurate estimates were provided by the EIS algorithm. This was also the case with respect to the mean MC bias, which was the smallest amongst the three methods. The second smallest MC bias was obtained with the Durbin-Koopman procedure, attaining the Davis-Yam algorithm the worst result in this respect. However, globally, the MC variance of the EIS algorithm was slightly worse than that resulting from the Durbin-Koopman method. In global terms, as measured by the Mean Squared Error (MSE), the most accurate η estimates were obtained with the Durbin-Koopman method, followed by the EIS algorithm. The worst MSE for these eleven parameters was attained with the Davis-Yam procedure. A different picture emerges with respect to the estimation of the semi-elasticities α, which are clearly the hardest parameters to estimate. Here results were somewhat mixed. For the majority of the α parameters the estimates with smaller MC bias were obtained with the EIS algorithm, followed by the Davis-Yam algorithm. However, this last method presented the worst average bias over the eleven α parameters for which there were at least 20 valid estimates. Nevertheless, the Durbin-Koopman presented, globally, the largest MC variance across the sensitivity parameters. This resulted in the less accurate estimates of the α parameters across the three different methods (as expressed by the MSE). The EIS algorithm was the method attaining the lowest MSE. Of particular importance is the bias and variance of the estimates of the latent autoregressive parameter ρ. Clearly the best estimates were obtained with the EIS algorithm. The Durbin-Koopman and Davis-Yam methods providing almost an equivalent level of accuracy (slightly better in the case of the DK method). The contrast between the results obtained with the DK method for the η and α parameters may indicate that this method is more effective at smoothing the state vector than estimating likelihood parameters. The simulation results for model B are presented in table 2.

<INSERT TABLE 2 ABOUT HERE>

Model B represents, clearly, the bigger challenge for the three estimation methods. The method that performed best overall, as assessed by the average MSE, was clearly the Durbin-Koopman approach. I start by looking at the estimates of the eleven η parameters. Here the EIS algorithm and the DK method performed similarly, in the sense that there were equal number of

cases where each method provided the most accurate MC mean estimate and the smallest MC variance. Nevertheless, the average MSE attained by the DK method over these parameters was smaller than that from the EIS algorithm. The Davis-Yam procedure performed poorly in both respects. With regard to the semi-elasticity parameters α, the EIS algorithm presented the lowest average MSE amongst the three methods. The DK method attained the second lowest average MSE. However, for most of these eleven parameters the lowest MC bias was achieved using this method, while the EIS algorithm provided the lowest MC variance in most of the cases. The Davis-Yam procedure generated the largest MC variance in most cases, the same being true globally. Again these patterns provide some evidence that the DK methodology performs better smoothing the state vector than providing estimates of the nonlinear model parameters. Particularly interesting were the results concerning the autoregressive parameter ρ. Despite performing worse for most of the remaining model parameters, when compared to model A, the EIS algorithm again delivered the smaller MC bias for the ρ parameter. However, the corresponding MC variance was the largest amongst the three methods. Nevertheless this resulted in a MSE almost identical to that obtained with the DK methodology. The Davis-Yam procedure performed particularly bad with regard to the MC bias of the ρ estimates (with low MC variance). This led to a very large MSE. Finally, Table 3 presents the results for model C.

<INSERT TABLE 3 ABOUT HERE>

Model C, presents perhaps the greatest computational challenge amongst the three models. The time-varying latent autoregressive coefficient and innovations variance leads to a slight increase in the complexity of the computations. This increased complexity in turn, can make the different algorithms slightly less robust to the accumulation of numerical errors. The most striking result, for this model, was the good performance of the Davis-Yam algorithm. The performance of this algorithm was particularly good when estimating the semi-elasticity parameters α. The average MSE (over the eleven α parameters) was the smallest amongst the three different methods, the same being true of the total squared MC bias. Additionally, for most of these parameters the MC bias and variance attained by this method were the smaller of the three methods. Looking at the η parameters, the EIS algorithm yielded the smaller average MSE. Simultaneously, this method was the most accurate and less variable across most of these parameters. Overall the best average MSE was delivered by the EIS algorithm, followed by the Davis-Yam procedure and then the Durbin-Koopman method. Finally the best estimates of the latent autoregressive coefficient ρ were provided by, in this order, the Davis-Yam procedure, the EIS algorithm and the Durbin-Koopman method.

<INSERT FIGURE 1 ABOUT HERE>

Finally, I present a comparison of the signal extraction algorithms associated with the three different estimation methods. In the first step I fix a representative simulated dataset generated under the set up of model C, jointly with the underlying (true) path of the latent process ψ. The next step consists in estimating both the model parameters and the posterior mean of the state vector using each of the three methods. By comparing the estimated path of the latent factor with the (true) underlying one, we can assess the relative performance of each method in recovering the latent process $\psi(t)$.

In figure 1 we can see the smoothed state vector (jointly with the corresponding 0.95 confidence bands) obtained from the Durbin-Koopman procedure, using equations (24) and (26). It is possible to see that the estimated (smoothed) path of the process $\psi(t)$ follows the true underlying values reasonably well. There are, however, some points where opposite local optima of both trajectories overlap (for example at moments $t = 5$, $t = 11$ and $t = 19$). The number of violations of the 0.95 confidence bands seems fairly reasonable.

<INSERT FIGURE 2 ABOUT HERE>

Figure 2 depicts the smoothed path of the latent process ψ obtained using the EIS algorithm. This can be obtained using formula (37) with the identity $\Phi(\mathbf{x}_N) \equiv \mathbf{x}_N$. Although 5 EIS iterations seem to yield satisfactory estimates of the model parameters (as documented by the MC results contained in tables 1, 2 and 3), the reliability of the corresponding estimates of the latent process ψ is very low. This is due to the fact that, typically, under a low number of EIS iterations, the resulting number of significant importance sampling weights[14] is low (usually only one or two). Not only this implies a high degree of variability for the resulting estimates of the posterior mean, but also that the estimates of the posterior variance are non-reliable (in the worst case, a value of zero is obtained). After severall unsuccessful attempts at obtaining (reasonable) smoothed estimates of the latent process using 5 EIS iterations I decided to increase this number to 25. Figures 2 and 4 were obtained with 25 EIS iterations. Further, I have investigated the speed of convergence of the EIS iterations as measured by convergence of the auxiliary regression parameters φ_n. In order to achieve an accuracy of at least 10^{-3} in φ_n approximately 90 EIS iterations were required. An accuracy of 10^{-5} for example, required approximately 156 EIS iterations. This is in sharp contrast with the results reported in Richard and Zhang (2007, sections 3.1 and 5) for several simpler models. Nevertheless, throughout all simulation experiments, convergence was always attained. The speed of convergence, however, seems to be very slow for the current target densities (18) when using Gaussian importance samplers. The question of whether it is possible to choose a different, more efficient, class of

[14] That is, importance sampling weights significantly larger than zero.

auxiliary samplers for the problem at hand is, however, outside of the scope of the current paper and is left for future research.

As it is possible to see in figure 2, using 25 EIS iterations of the Gaussian importance samplers, the smoothed estimates of the latent process ψ are very accurate. The posterior mode of ψ follows closely the true underlying values. The 0.95 confidence bands are very informative, clearly identifying periods of large up and down movements of the latent process. The number of violations is reasonable. The clear down-side of using an increased number of EIS iterations is the required computational time. Obtaining the results reported in figure 2, required more than two hours in a desktop PC built around an Intel Pentium IV processor running at 3.4 GHz under Windows XP Professional SP2.

<INSERT FIGURE 3 ABOUT HERE>

As seen in section 4, the approximate Gaussian density $p_a(\mathbf{x}_N|\mathbf{z}_N;\theta)$ is interpretable as an importance density. Therefore we can obtain smoothed estimates of the state process by using formula (50). Figure 3 presents the resulting path of the latent process ψ jointly with its posterior mode (i.e. the mode of p_a). The confidence bands were obtained using the posterior variance of the approximate posterior density p_a, and can be used to assess (jointly with the difference between the posterior mode and mean of ψ) whether or not the (exact) posterior density of the latent process ψ reasonably resembles a Gaussian density. The answer seems negative. In fact not only the confidence bands are reasonably wide, but there is also a noticeable difference between the posterior mode and the (estimated) posterior mean (in sharp contrast with the results for the SV model reported in figures 5 and 6 in Davis and Rodriguez-Yam, 2005). Nevertheless the smoothed path of the latent process follows fairly well the true values. The number of violations of the 0.95 confidence bands is low. Additionally, the local optima from both trajectories coincides frequently. As it is to expect, the posterior mode excessively smoothes the true path of the latent process.

<INSERT FIGURE 4 ABOUT HERE>

Finally, figure 4 directly compares the output of the three different smoothing procedures. The smoothed estimates of the latent process (jointly with the posterior mode from the DY procedure) are plotted against the true trajectory. In this graphic the confidence bands are omitted to enhance visibility. The main feature of this figure is the high accuracy of the EIS smoothed estimate of the latent process. The EIS estimate of the posterior mean of ψ follows closely the true trajectory. The posterior mode of ψ obtained from the DY procedure presents an excessive degree of smoothing. The DY estimate of the posterior mean, on the other hand,

overestimates the true values between periods $t = 1$ and $t = 10$, and underestimates between periods $t = 21$ and $t = 24$. The DK procedure provides reasonably accurate smoothed estimates of the latent process. Nevertheless, there are some noticeable errors, for example around periods $t = 5$, $t = 11.5$ and $t = 19.5$.

6 Conclusion

This paper reviewed three feasible methodologies for conducting maximum likelihood estimation of parameter driven dynamic statistical models. I described in detail how to apply each method to the class of Multi-state Latent Factor Intensity models, introduced in Koopman et al. (2008). A Monte Carlo study was conducted to assess the behaviour of these estimation methods when applied to a relatively complex econometric model.

Two of the methods reviewed require the use of Monte Carlo integration with importance sampling techniques. In order to construct the importance density, one of these methods, the EIS algorithm, requires solving recursively a sequence of low-dimensional least squares problems. Implementing this procedure is relatively simple. Furthermore, the simulation study revealed that this method provides very accurate parameter estimates. However, this method is computationally heavy. It was always the slower of the three methods under study. A high number of iterations of the basic steps of the algorithm is required for yielding a satisfactory level of accuracy, in particular for signal extraction purposes. The other simulation based approach, the Durbin-Koopman methodology, requires constructing a linear Gaussian model for the complete data (i.e. observed data plus state variables). This is somewhat complex, as this is done by solving iteratively a high-dimensional nonlinear equation, using at every step the Kalman filtering and smoothing recursions associated with the approximating Gaussian model. Nevertheless, the resulting estimation and signal extraction procedures appear to be computationally efficient. The third method avoids the need for using Monte Carlo integration by resorting to a closed-form approximation to the loglikelihood function implied by the model. This led, in some cases, to a noticeable Monte Carlo bias. The Monte Carlo squared bias and variance of the DK methodology, in general terms, were larger than their EIS counterparts. In this respect the closed-form Likelihood approximation of Davis and Rodriguez-Yam (2005) seems to yield the larger estimation errors. However, overall results were mixed. Furthermore, the signal extraction procedure directly derived from the DY method delivers reasonable estimates of the latent process.

Finally, this paper shows that, although relatively complex and computationally demanding, the estimation of Multi-state Latent Factor Intensity Models is clearly feasible, even when there

is a large number of parameters. This opens the way for other applications of the framework, not necessarily confined to the analysis of Agency credit rating data.

Particular thanks are due to my thesis Advisor Prof. André Lucas, and to Prof. Siem Jan Koopman, for their assistance with the implementation of the Durbin-Koopman methodology and providing the corresponding `Ox` code. I further wish to thank Prof. Koopman for kindly providing a copy of the `Ox` package `SsfPack`. Finally, I wish to acknowledge the contribution to this work that resulted from the helpful discussions with my thesis Advisor. All calculations were performed using the `Ox` matrix programming language of Doornik (2002). The Durbin-Koopman methodology was implemented using the routines contained in the `Ox` package `SsfPack` (version beta 3.05) of Doornik, Koopman, and Shephard (1998). I wish to thank Prof. Michael McAleer, Prof. David Allen and Dr. Zhaoyong Zhang for helpful comments. Financial support by the Australian Research Council is gratefully acknowledged.

References

Andersen, P.K., Borgan, Ø., Gill, R.D., and N. Keiding, (1993): *Statistical models based on counting processes*. Springer-Verlag, New York.

Bauwens, L., and D. Veredas, (2004): "The stochastic conditional duration model: a latent factor model for the analysis of financial durations," *Journal of Econometrics* **119**, No. 2, pp. 381-412.

Bauwens, L. and N. Hautsch (2006): "Stochastic conditional intensity processes," *Journal of Financial Econometrics* **4**, No. 3, pp. 450-493.

Bowsher, C. G. (2007): "Modelling security market events in continuous time: intensity based, multivariate point process models," *Journal of Econometrics* **141**, pp. 876-912.

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

Brown, T.C. and Nair, M.G. (1988): "A simple proof of the multivariate time change theorem for point processes," *Journal of Applied Probability* **25**, pp. 210-214.

Cox, D.R. (1962): *Renewal Theory*, Methuen and Co. Ltd.

Davis, R.A. and G.R. Yam, (2005): "Estimation for state-space models: an approximate likelihood approach," *Statistica Sinica* **15**, pp. 381-406.

Duffie, D., A. Eckner, G. Horel, and L. Saita (2006): "Frailty correlated default," *Working paper - Stanford University*.

Duffie, D., L. Saita, and K. Wang (2007): "Multi-period corporate default prediction with stochastic covariates," *Journal of Financial Economics*, **83**, pp. 635-665.

Durbin, J. and S.J. Koopman, (1997): "Monte Carlo maximum likelihood estimation for non-Gaussian state space models," *Biometrika* **84**, pp. 669-684.

Durbin, J. and S.J. Koopman, (2001): *Time Series Analysis by State Space Methods*. Oxford University Press, Oxford.

Durbin, J. and S.J. Koopman, (2002): "A simple and efficient simulation smoother for state space time series models," *Biometrika* **89**, pp. 603-616.

Engle, R.F. and J.R. Russell, (1998): "Autoregressive conditional duration: a new model for irregularly spaced transaction data," *Econometrica* **66**, No. 5, pp. 1127-1162.

Glynn, P.W. (1988): "A GSMP formalism for discrete-event systems," *Proceedings of the IEEE* **77**, pp. 14-23.

Harvey, A.C. (1989): *Forecasting, Structural Time Series Models and the Kalman Filter*. Cambridge University Press, Cambridge.

Hasbrouck, J. (1991): "Measuring the information content of stock trades," *The Journal of Finance*, **46**, No. 1 pp. 179-207.

Kavvathas, D. (2001): "Estimating credit rating transition probabilities for corporate bonds," *Working Paper - University of Chicago*

Koopman, S.J., Lucas, A. and A.A. Monteiro (2008): "The multi-state latent factor intensity model for credit rating transitions," *Journal of Econometrics* **142** No. 1, pp. 405-430.

Koopman S.J., R. Kraeussl, A. Lucas, and A.A. Monteiro, (2006): " Credit cycles and macro fundamentals," *CFS Working Paper Nr. 2006/33 University of Frankfurt.*

Lando, D. and T.M. Skødeberg (2002): "Analyzing rating transitions and rating drift with continuous observations," *Journal of Banking and Finance*, **6**, pp. 423-444.

Liesenfeld, R. & J.F. Richard (2003): "Univariate and multivariate stochastic volatility models: estimation and diagnostics," *Journal of Empirical Finance* **10**, pp. 505-531.

Richard, J-F. and W. Zhang. (2007): "Efficient high-dimensional importance sampling," *Journal of Econometrics*, **141**, pp. 1385-1411.

Russell, J.R. (1999): "Econometric modelling of multivariate irregularly spaced high-frequency data," *Working Paper - University of Chicago.*

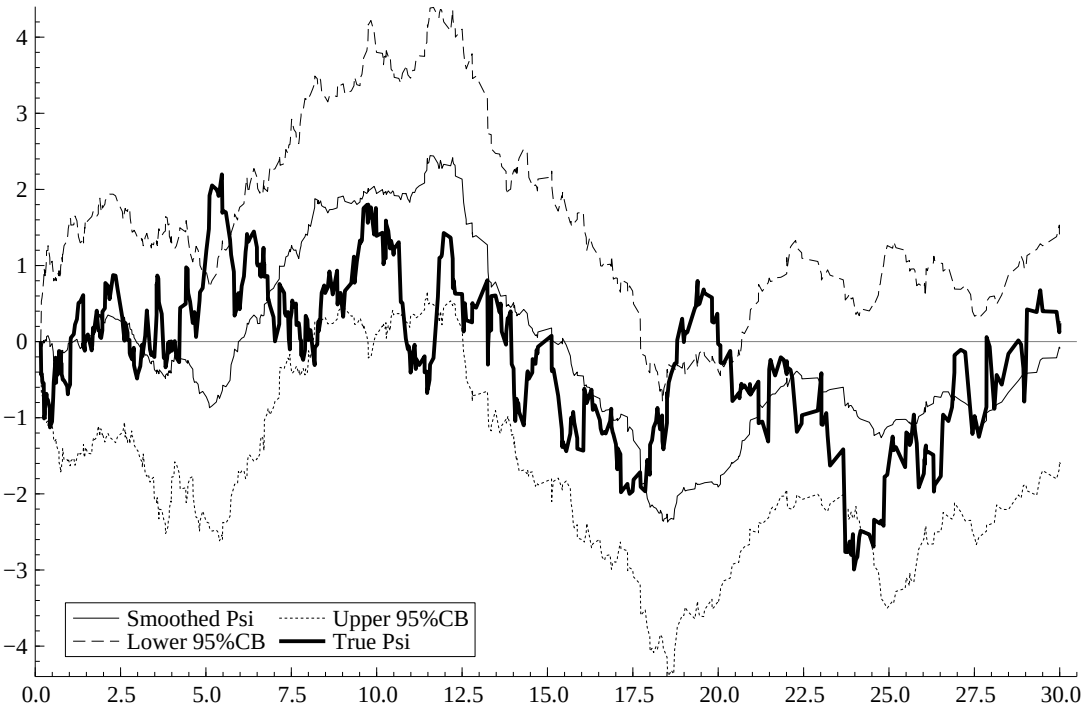

Figure 1: Signal Extraction: Durbin-Koopman Method

This graphic illustrates the signal extraction capabilities of the Durbin-Koopman SML methodology, when applied to the MLFI model

$$\lambda_s^k(t) = Y_s^k(t) \cdot \exp[\eta_s + \alpha_s \psi(t)],$$

for $k = 1, \ldots, K$ with the number of units $K = 100$, and the number of transition types $s = 1, \ldots, 16$. A simulated dataset was generated over a time-window of length 30. The precise simulation set up is explained in section 5. The horizontal axis represents (calendar) time. There is a single latent discretized O-U process ψ (the details of this specification are presented in the text). The true parameters of the model are presented in the first column of table 3. The thick solid line is the true path of the underlying ψ variable, while the thin solid line represents the estimated values.

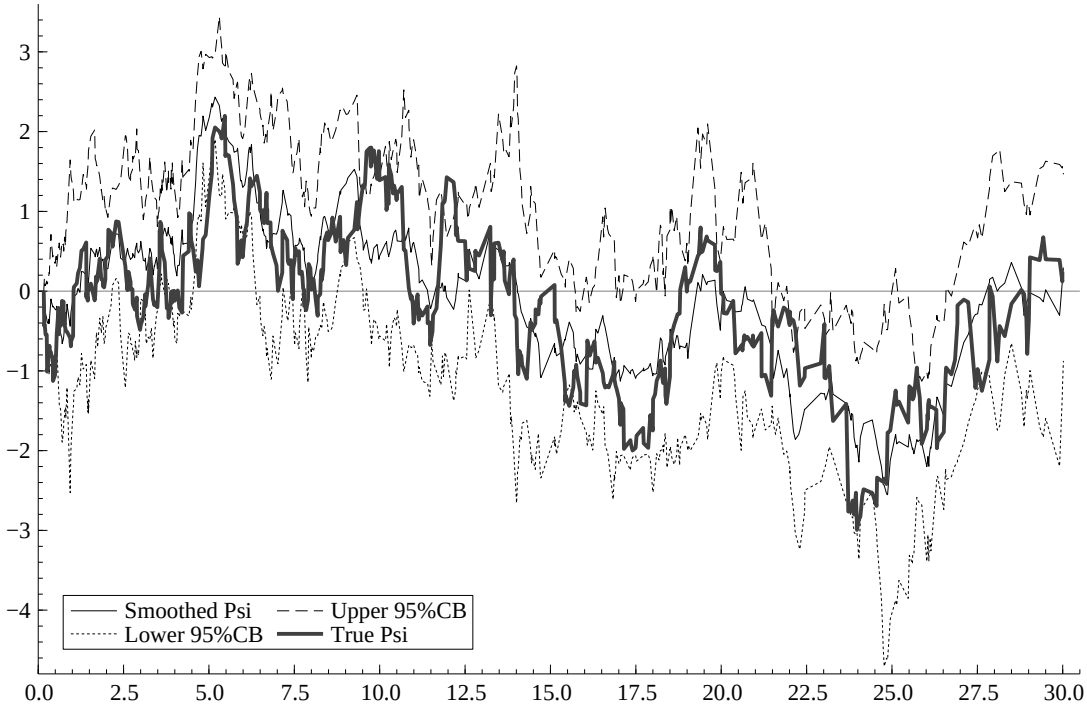

Figure 2: Signal Extraction: EIS Algorithm

This graphic illustrates the signal extraction capabilities of the EIS algorithm, when applied to the MLFI model

$$\lambda_s^k(t) = Y_s^k(t) \cdot \exp[\eta_s + \alpha_s \psi(t)],$$

for $k = 1, \ldots, K$ with the number of units $K = 100$, and the number of transition types $s = 1, \ldots, 16$. A simulated dataset was generated over a time-window of length 30. The precise simulation set up is explained in section 5. The horizontal axis represents (calendar) time. There is a single latent discretized O-U process ψ (the details of this specification are presented in the text). The true parameters of the model are presented in the first column of table 3. The thick solid line is the true path of the underlying ψ variable, while the thin solid line represents the estimated values.

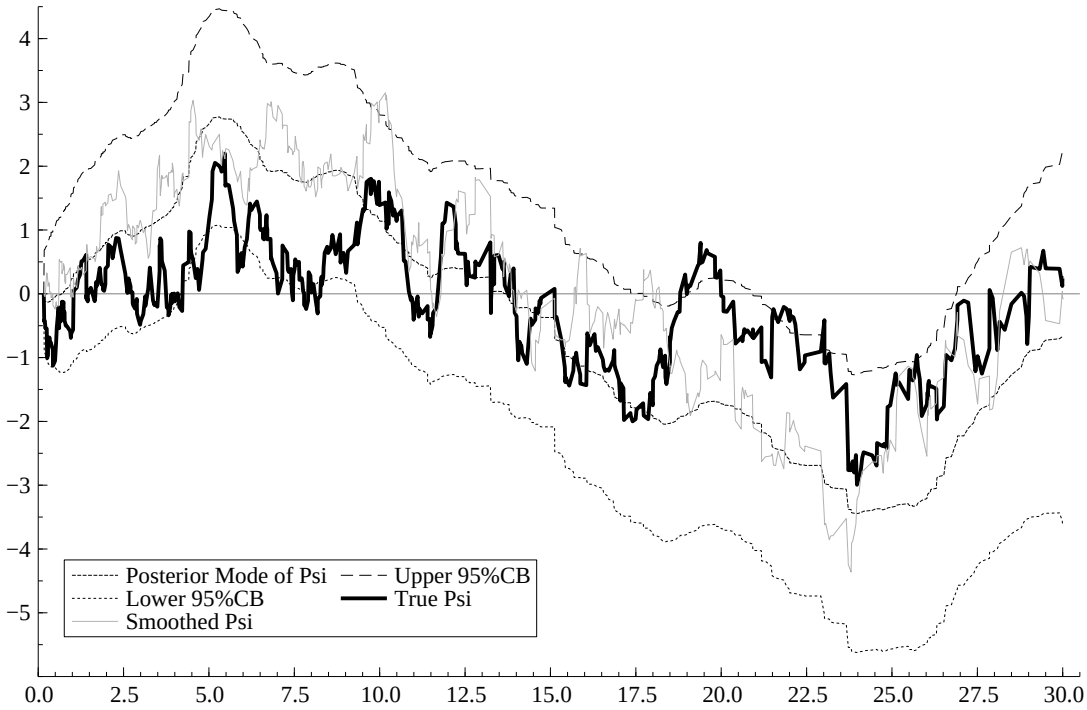

Figure 3: Signal Extraction: Approximate ML Method

This graphic illustrates the signal extraction capabilities of the Davis-Yam approximate ML procedure, when applied to the MLFI model

$$\lambda_s^k(t) = Y_s^k(t) \cdot \exp[\eta_s + \alpha_s \psi(t)],$$

for $k = 1, \ldots, K$ with the number of units $K = 100$, and the number of transition types $s = 1, \ldots, 16$. A simulated dataset was generated over a time-window of length 30. The precise simulation set up is explained in section 5. The horizontal axis represents (calendar) time. There is a single latent discretized O-U process ψ (the details of this specification are presented in the text). The true parameters of the model are presented in the first column of table 3. The thick solid line is the true path of the underlying ψ variable, while the thin solid line represents the estimated values.

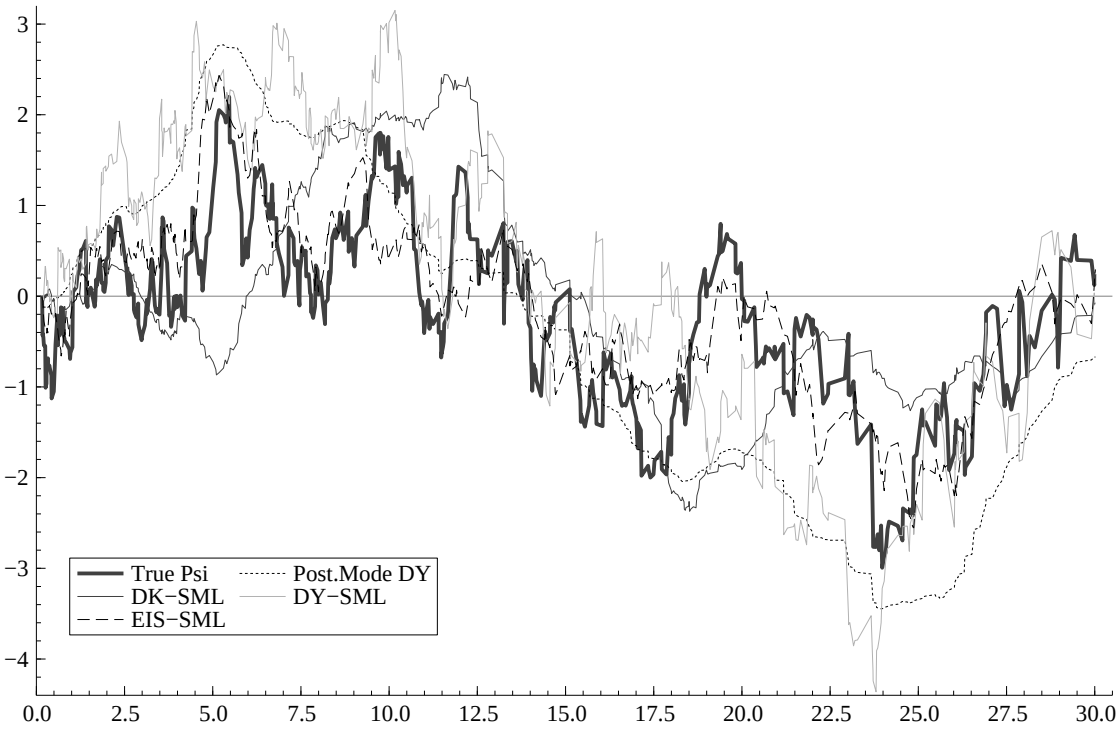

Figure 4: Comparison of Signal Extraction algorithms

This picture presents a direct comparison between the signal extraction capabilities of the three different estimation methods covered in this paper, when applied to the MLFI model

$$\lambda_s^k(t) = Y_s^k(t) \cdot \exp[\eta_s + \alpha_s \psi(t)],$$

for $k = 1, \ldots, K$ with the number of units $K = 100$, and the number of transition types $s = 1, \ldots, 16$. There is a single latent discretized O-U process ψ (the details of this specification are presented in the text). The true parameters of the model are presented in the first column of table 3. The three different estimation algorithms were applied to the same simulated dataset (DK-SML indicates the Durbin-Koopman SML methodology, EIS-SML corresponds to the SML technique of Liesenfeld and Richard and DY-SML denotes the simulation-based smoothing procedure derived from the Davis-Yam approximate ML algorithm). The thick solid line is the true path of the underlying ψ variable. The precise simulation set up is explained in section 5.

Table 1: Monte Carlo Results I: persistent AR(1)

This table contains parameter estimates for the MLFI model, $\lambda_s^k(t) = Y_s^k(t) \cdot \exp[\eta_s + \alpha_s \psi(t)]$, for $k = 1, \ldots, K$ with the number of units $K = 100$, and the number of transition types $s = 1, \ldots, 16$. The true parameters are presented in the first column. There are 4 rating classes (plus default). Initial ratings are distributed evenly over these classes. Here I estimate the model with a single latent stationary AR(1) process ψ, with AR parameter $\rho = 0.9$. The maximum time T is set to 30 time-units, unless the complete sample has entered into the absorbing (default) state at an earlier stage. I performed 100 replications for each estimation method using the same simulated datasets (DK-SML indicates the Durbin-Koopman SML methodology, EIS-SML corresponds to the SML technique of Liesenfeld and Richard and DY-AML denotes the Davis-Yam approximate ML procedure). Superscript numbers between parentheses, in the column containing the mean values, indicate the number of valid estimates used to compute the Monte Carlo statistics over all simulations.

Parameter	True	DK-SML		EIS-SML		DY-AML	
	Value	Mean	Std.Error	Mean	Std.Error	Mean	Std.Error
$\eta_{1\to2}$	-3.47	-3.55	(0.20)	-3.47	(0.16)	-3.51	(0.13)
$\eta_{1\to3}$	-5.88	$-6.25^{(96)}$	(0.80)	$-6.11^{(96)}$	(0.89)	$-6.20^{(96)}$	(0.86)
$\eta_{2\to1}$	-5.04	-5.18	(0.49)	-5.13	(0.39)	-5.72	(1.50)
$\eta_{2\to3}$	-3.04	-3.17	(0.21)	-3.06	(0.14)	-3.09	(0.14)
$\eta_{2\to4}$	-5.84	$-6.32^{(97)}$	(0.82)	$-5.94^{(97)}$	(0.50)	$-6.06^{(97)}$	(0.95)
$\eta_{2\to D}$	-7	$-7.03^{(25)}$	(1.12)	$-6.96^{(25)}$	(0.44)	$-6.52^{(25)}$	(0.35)
$\eta_{3\to2}$	-4.5	-4.58	(0.49)	-4.65	(0.38)	-5.09	(1.47)
$\eta_{3\to4}$	-3.1	-3.19	(0.19)	-3.13	(0.13)	-3.14	(0.12)
$\eta_{3\to D}$	-2.5	-2.61	(0.16)	-2.52	(0.13)	-2.54	(0.11)
$\eta_{4\to3}$	-5.2	$-5.77^{(91)}$	(1.48)	$-5.96^{(91)}$	(2.40)	$-6.76^{(91)}$	(3.60)
$\eta_{4\to D}$	-1.7	-1.69	(0.24)	-1.75	(0.24)	-1.84	(0.24)
$\alpha_{1\to3}$	-0.1	$-0.12^{(96)}$	(0.24)	$-0.22^{(96)}$	(0.31)	$-0.46^{(96)}$	(1.02)
$\alpha_{2\to1}$	0.1	$-0.11^{(56)}$	(1.20)	0.15	(0.20)	0.63	(1.32)
$\alpha_{2\to3}$	-0.2	-0.14	(0.43)	-0.24	(0.09)	-0.22	(0.12)
$\alpha_{2\to4}$	-0.08	$-0.26^{(97)}$	(0.26)	$-0.17^{(97)}$	(0.23)	$-0.26^{(97)}$	(0.78)
$\alpha_{2\to D}$	-0.01	$0.05^{(25)}$	(0.61)	$-0.17^{(25)}$	(0.26)	$-0.12^{(25)}$	(0.21)
$\alpha_{3\to2}$	0.12	$0.68^{(75)}$	(2.15)	0.18	(0.16)	0.63	(1.34)
$\alpha_{3\to4}$	-0.1	-0.18	(0.40)	-0.12	(0.09)	-0.11	(0.11)
$\alpha_{3\to D}$	-0.2	-0.13	(0.26)	-0.23	(0.07)	-0.20	(0.12)
$\alpha_{4\to3}$	0.1	$0.56^{(56)}$	(1.74)	$0.30^{(91)}$	(0.38)	$0.86^{(91)}$	(1.51)
$\alpha_{4\to D}$	-0.5	-0.26	(0.71)	-0.56	(0.11)	-0.65	(0.33)
ρ	0.90	0.79	(0.20)	0.87	(0.03)	0.77	(0.24)

Table 2: Monte Carlo Results II: transitory AR(1)

This table contains parameter estimates for the MLFI model, $\lambda_s^k(t) = Y_s^k(t) \cdot \exp[\eta_s + \alpha_s \psi(t)]$, for $k = 1, \ldots, K$ with the number of units $K = 100$, , and the number of transition types $s = 1, \ldots, 16$. The true parameters are presented in the first column. There are 4 rating classes (plus default). Initial ratings are distributed evenly over these classes. Here I estimate the model with a single latent stationary AR(1) process ψ, with a somewhat low AR parameter of $\rho = 0.5$. The maximum time T is set to 30 time-units, unless the complete sample has entered into the absorbing (default) state at an earlier stage. I performed 100 replications for each estimation method using the same simulated datasets (DK-SML indicates the Durbin-Koopman SML methodology, EIS-SML corresponds to the SML technique of Liesenfeld and Richard and DY-AML denotes the Davis-Yam approximate ML procedure). Monte-Carlo averages and standard errors (in parentheses) are presented for those parameters that have a sufficient number of occurrences over all simulations. Superscript numbers between parentheses, in the column containing the mean values, indicate the number of valid estimates used to compute both Monte Carlo statistics over all simulations.

Parameter	True	DK-SML		EIS-SML		DY-AML	
	Value	Mean	Std.Error	Mean	Std.Error	Mean	Std.Error
$\eta_{1\to2}$	-3.47	-3.56	(0.21)	-3.48	(0.14)	-3.63	(0.67)
$\eta_{1\to3}$	-5.88	$-6.18^{(94)}$	(0.70)	$-6.30^{(94)}$	(0.93)	$-6.66^{(94)}$	(1.59)
$\eta_{2\to1}$	-5.04	-5.20	(0.49)	-5.39	(0.74)	-6.78	(2.20)
$\eta_{2\to3}$	-3.04	-3.09	(0.16)	-3.06	(0.11)	-3.18	(0.28)
$\eta_{2\to4}$	-5.84	$-6.15^{(96)}$	(0.72)	$-6.09^{(96)}$	(0.64)	$-6.72^{(96)}$	(1.87)
$\eta_{2\to D}$	-7	$-7.20^{(32)}$	(1.03)	$-7.96^{(32)}$	(3.00)	$-7.75^{(32)}$	(2.13)
$\eta_{3\to2}$	-4.5	-4.63	(0.52)	-4.99	(0.80)	-7.22	(2.14)
$\eta_{3\to4}$	-3.1	-3.22	(0.24)	-3.14	(0.13)	-3.39	(0.93)
$\eta_{3\to D}$	-2.5	-2.57	(0.16)	-2.51	(0.10)	-2.61	(0.24)
$\eta_{4\to3}$	-5.2	$-5.65^{(85)}$	(1.00)	$-6.16^{(85)}$	(1.52)	$-6.32^{(85)}$	(1.92)
$\eta_{4\to D}$	-1.7	-1.70	(0.12)	-1.72	(0.10)	-1.83	(0.19)
$\alpha_{1\to3}$	-0.1	$-0.09^{(94)}$	(0.44)	$-0.50^{(94)}$	(0.72)	$-0.01^{(76)}$	(0.06)
$\alpha_{2\to1}$	0.1	$-0.09^{(71)}$	(0.97)	0.41	(0.63)	0.49	(1.34)
$\alpha_{2\to3}$	-0.2	-0.12	(0.61)	-0.28	(0.27)	-0.27	(0.62)
$\alpha_{2\to4}$	-0.08	$-0.20^{(96)}$	(0.36)	$-0.41^{(96)}$	(0.63)	$-0.03^{(77)}$	(0.08)
$\alpha_{2\to D}$	-0.01	$-0.23^{(32)}$	(0.81)	$-0.56^{(32)}$	(0.92)	$-0.03^{(23)}$	(0.07)
$\alpha_{3\to2}$	0.12	$0.26^{(71)}$	(1.44)	0.52	(0.69)	$0.44^{(41)}$	(1.27)
$\alpha_{3\to4}$	-0.1	-0.11	(0.60)	-0.24	(0.31)	$-0.16^{(96)}$	(0.80)
$\alpha_{3\to D}$	-0.2	-0.23	(0.40)	-0.23	(0.21)	-0.17	(0.47)
$\alpha_{4\to3}$	0.1	$0.11^{(64)}$	(0.93)	0.63	(0.86)	$0.42^{(70)}$	(1.23)
$\alpha_{4\to D}$	-0.5	-0.47	(0.94)	-0.46	(0.23)	-0.57	(0.57)
ρ	0.50	0.56	(0.26)	0.46	(0.28)	0.06	(0.12)

Table 3: Monte Carlo Results III: discretized O-U process

This table contains parameter estimates for the MLFI model, $\lambda_s^k(t) = Y_s^k(t) \cdot \exp[\eta_s + \alpha_s \psi(t)]$, for $k = 1, \ldots, K$ with the number of units $K = 100$, , and the number of transition types $s = 1, \ldots, 16$. The true parameters are presented in the first column. There are 4 rating classes (plus default). Initial ratings are distributed evenly over these classes. Here I estimate the model with a single latent discretized O-U process ψ (the details of this specification are presented in the text), with parameter $\rho = 0.9$. The maximum time T is set to 30 time-units, unless the complete sample has entered into the absorbing (default) state at an earlier stage. I performed 100 replications for each estimation method using the same simulated datasets (DK-SML indicates the Durbin-Koopman SML methodology, EIS-SML corresponds to the SML technique of Liesenfeld and Richard and DY-AML denotes the Davis-Yam approximate ML procedure) as far as possible.[a] Monte-Carlo averages and standard errors (in parentheses) are presented for those parameters that have a sufficient number of occurrences over all simulations. Superscript numbers between parentheses indicate the number of valid estimates used to compute the Monte Carlo statistics over all simulations.

[a] In 4 out of the 100 common simulated datasets the `Ox` implementation of the DY-AML method collapsed numerically (providing no estimates). In these cases estimation was performed over 4 distinct simulated datasets.

Parameter	True	DK-SML		EIS-SML		DY-AML	
	Value	Mean	Std.Error	Mean	Std.Error	Mean	Std.Error
$\eta_{1\to2}$	-3.47	-3.49	(0.29)	-3.44	(0.16)	-3.54	(0.22)
$\eta_{1\to3}$	-5.88	$-6.47^{(93)}$	(2.15)	$-5.92^{(93)}$	(0.43)	$-6.18^{(93)}$	(0.64)
$\eta_{2\to1}$	-5.04	-5.27	(0.56)	-5.15	(0.32)	-5.05	(0.31)
$\eta_{2\to3}$	-3.04	-3.12	(0.42)	-2.97	(0.26)	-3.19	(0.18)
$\eta_{2\to4}$	-5.84	-6.17	(1.42)	-6.13	(1.70)	-6.33	(2.56)
$\eta_{2\to D}$	-7	$-6.66^{(35)}$	(1.40)	$-6.53^{(35)}$	(0.63)	$-6.62^{(35)}$	(0.42)
$\eta_{3\to2}$	-4.5	-4.72	(0.64)	-4.58	(0.42)	-4.47	(0.33)
$\eta_{3\to4}$	-3.1	-3.22	(0.52)	-3.11	(0.19)	-3.22	(0.19)
$\eta_{3\to D}$	-2.5	-2.53	(0.32)	-2.44	(0.24)	-2.65	(0.19)
$\eta_{4\to3}$	-5.2	$-5.32^{(78)}$	(0.95)	$-5.11^{(78)}$	(0.44)	$-5.15^{(78)}$	(0.79)
$\eta_{4\to D}$	-1.7	-1.58	(0.45)	-1.52	(0.62)	-2.04	(0.34)
$\alpha_{1\to2}$	-0.08	-0.079	(0.004)	-0.05	(0.08)	-0.09	(0.11)
$\alpha_{1\to3}$	-0.1	$-0.08^{(93)}$	(0.19)	$-0.18^{(93)}$	(0.31)	$-0.23^{(93)}$	(0.29)
$\alpha_{2\to1}$	0.1	-0.033	(2.01)	0.11	(0.17)	0.15	(0.20)
$\alpha_{2\to3}$	-0.2	-0.13	(0.36)	-0.07	(0.10)	-0.19	(0.09)
$\alpha_{2\to4}$	-0.08	-0.18	(0.15)	-0.16	(0.27)	-0.17	(0.25)
$\alpha_{2\to D}$	-0.01	$-0.11^{(35)}$	(0.41)	$-0.19^{(35)}$	(0.34)	$-0.19^{(35)}$	(0.30)
$\alpha_{3\to2}$	0.12	0.29	(2.23)	0.19	(0.34)	0.17	(0.23)
$\alpha_{3\to4}$	-0.1	-0.09	(0.42)	-0.07	(0.11)	-0.13	(0.12)
$\alpha_{3\to D}$	-0.2	-0.15	(0.20)	-0.08	(0.11)	-0.20	(0.09)
$\alpha_{4\to3}$	0.1	$0.31^{(78)}$	(1.11)	$0.14^{(78)}$	(0.23)	$0.29^{(78)}$	(0.65)
$\alpha_{4\to D}$	-0.5	-0.27	(0.81)	-0.18	(0.46)	-0.49	(0.14)
ρ	0.90	0.73	(0.28)	0.75	(0.29)	0.79	(0.22)