ESTIMATION OF DYNAMIC LATENT VARIABLE MODELS USING SIMULATED NONPARAMETRIC MOMENTS

MICHAEL CREEL

Universitat Autònoma de Barcelona

June 2008

ABSTRACT. Given a model that can be simulated, conditional moments at a trial parameter value can be calculated with high accuracy by applying kernel smoothing methods to a long simulation. With such conditional moments in hand, standard method of moments techniques can be used to estimate the parameter. Because conditional moments are calculated using kernel smoothing rather than simple averaging, it is not necessary that the model be simulable subject to the conditioning information that is used to define the moment conditions. For this reason, the proposed estimator is applicable to general dynamic latent variable models. The estimator is consistent and has the same asymptotic distribution as that of the infeasible GMM estimator based on the same moment conditions. Monte Carlo results show how the estimatod may be applied to a range of dynamic latent variable (DLV) models, and that it performs well in comparison to several other estimators that have been proposed for DLV models. An application to weekly spot exchange rate data further illustrates use of the estimator.

Keywords: dynamic latent variable models; simulation-based estimation; simulated moments; kernel regression; nonparametric estimation

JEL codes: C13; C14; C15

Department of Economics and Economic History, Edifici B, Universitat Autònoma de Barcelona, 08193 Bellaterra (Barcelona) Spain. michael.creel@uab.es This research was supported by grants SEJ 2006-00712/ECON and CONSOLIDER-INGENIO 2010 (CSD2006-00016).

1. INTRODUCTION

Dynamic latent variable (DLV) models are a flexible and often natural way of modeling complex phenomena. As an example, consider a macroeconomic model. A model may specify behavioral rules, learning rules, a social networking structure, and information transmission mechanisms for a large group of possibly heterogeneous agents. If the model is fully specified, it can be used to generate time series data on all of the agents' actions. In attempting to use real world data to estimate the parameters of such model, one finds that real world data is much more aggregated than the data generated by the model. Typically, individual agents' actions are not observed - only macroeconomic aggregates are available. From the econometric point of view, many of the variables generated by the model are latent. In a dynamic, nonlinear context, this can complicate the econometric estimation of the model's parameters.

To fix ideas, consider the general DLV model:

$$\text{DLV: } \begin{cases} y_t = & r_t\left(y^{t-1}, y^{*t}, \varepsilon_t; \theta\right) \\ y_t^* = & r_t^*\left(y^{t-1}, y^{*t-1}, \varepsilon_t; \theta\right) \end{cases} \tag{1}$$

where $t = 1, ..., n$. The observable variables are the k_y dimensional vector y_t, and y_t^* is a vector of latent variables. Superscript notation is used to indicate the entire history of a vector up to the time indicated, so $y^{t-1} \equiv \left(y_1', ..., y_{t-1}'\right)'$, and $y^{*t-1} \equiv \left(y_1^{*\prime}, ..., y_{t-1}^{*\prime}\right)'$. There is a vector of independent white noises, ε_t, with a known distribution. Finally, θ is a vector of unknown parameters[1]. This definition closely follows that of Billio and Monfort (2003), with the exception that the same white noise vector enters the equations for both the observable and latent variables, to allow for potential correlations in the innovations of the two sets of variables. Calculation of the likelihood function requires finding the density of y^n, and as Billio and Monfort make clear, this involves calculating an integral of the same order as n, a problem that is in general untractable. Without the density of the observable variables, analytic moments cannot be computed. Thus, maximum likelihood and moment-based estimation methods often are not available.

A number of econometric methods have been developed over the last two decades to deal with the complications that may accompany DLV models. These include the simulated method of moments (McFadden, 1989; Pakes and Pollard, 1989), indirect inference (Gouriéroux, Monfort and Renault, 1993; Smith, 1993), simulated pseudo-maximum likelihood (Laroque and Salanié, 1993), simulated maximum likelihood (Lee, 1995), the efficient method of moments (Gallant and Tauchen, 1996), the method of simulated scores (Hajivassiliou and McFadden, 1998), kernel-based indirect inference (Billio and Monfort, 2003), the simulated EM algorithm (Fiorentini, Sentana and Shephard, 2004), nonparametric simulated maximum likelihood (Fermanian and Salanié, 2004; Kristensen and Shin, 2006) and simulated nonparametric estimators (Altissimo and Mele, 2007). These methods have been

[1] The possible presence of observable exogenous variables is suppressed for clarity. The macroeconomic model of the previous paragraph could be formalized by letting y_t^* indicate the vector of all of the agents' actions, and letting y_t be the observed aggregate outcomes.

applied to DLV models in a number of contexts. Billio and Monfort (2003) provide numerous references for applications.

As noted by Fermanian and Salanié (2004, pg. 702), there often exists a trade-off between the asymptotic efficiency of a method and its applicability to a wide range of models. Simulated maximum likelihood and the method of simulated scores are asymptotically efficient when they can be applied, but this is not the case when the likelihood function or the score function cannot be expressed as a function of expectations of simulable quantities. Nonparametric simulated maximum likelihood (NPSML) is asymptotically efficient and generally applicable for estimation of static models (Fermanian and Salanié, 2004). Kristensen and Shin (2006) extend the method to some dynamic models. In general, the method encounters curse-of-dimensionality problems in the case of dynamic models. Proposed solutions based upon lower dimensional marginals of the likelihood function lead to a loss of asymptotic efficiency.

The simulated method of moments (SMM) is generally applicable if unconditional moments are used, but foregoing conditioning information may limit the estimator's ability to capture the dynamics of the model, and can result in poor efficiency (Andersen, Chung and Sorensen, 1999; Michaelides and Ng, 2000; Billio and Monfort, 2003). In the context of DLV models, the usual implementation of SMM that directly averages a simulator normally cannot be based upon conditional moments, since it is not in general possible to simulate from the model subject to the conditioning information. Due to the full specification of the model, it is easy to simulate a path, $\widetilde{y^n}(\theta)$. However, the elements are drawn from their marginal distributions. It is not in general possible to draw from $y_t|y^{t-1};\theta$. To do so, one would need draws from $y^{*t}|y^{t-1};\theta$. If such draws were available, they could be inserted into the first line of the DLV model given in equation 1, which, combined with a draw from ε_t, would give a draw from $y_t|y^{t-1};\theta$. The problem is that the observed value of y^{t-1} is only compatible with certain realizations of the history of the latent variables, y^{*t-1}, but what is the set of compatible realizations is not known. For certain types of model it is possible to circumvent this problem. For example, Fiorentini, Sentana and Shephard (2004) find a way of casting a factor GARCH model as a first-order Markov process, and are then able to use Markov chain Monte Carlo (MCMC) methods to simulate from $y^{*t}|y^{t-1};\theta$, which is then fed into a simulated EM algorithm to estimate the parameter. However, for DLV models in general, there is no means of simulating from $y^{*t}|y^{t-1};\theta$ (Billio and Monfort, 2003, pg. 298; Carrasco *et al.*, 2007, pg. 544).

Indirect inference is generally applicable, but its efficiency depends crucially upon the choice of the auxiliary model. The efficient method of moments (EMM, Gallant and Tauchen, 1996) is closely related to the indirect inference estimator, and presumes use of an auxiliary model that guarantees good asymptotic efficiency, by closely approximating the structural model. This estimator is both generally applicable and is highly efficient if a good auxiliary model is used, and it is fully asymptotically efficient if the auxiliary model satisfies a smooth embedding condition (see Gallant and Tauchen, 1996, Definition 1). Satisfying this condition is not necessarily an easy thing to achieve. A common practice is to fit a semi-nonparametric (SNP) auxiliary model of the sort proposed by Gallant and Nychka (1987), augmented by a leading parametric model that is known to provide a reasonably good approximation. Andersen, Chung and Sorensen (1999) provide Monte Carlo evidence that shows the importance of the choice of the auxiliary model. They also note that highly parameterized auxiliary models often cannot be successfully fit when the sample size is not large. It is important to keep in mind that a parsimonious parametric auxiliary model may be far from satisfying the smooth embedding condition. This can lead to serious inefficiency and to failure to detect serious misspecifications of the structural model (Tauchen, 1997; Gallant and Tauchen, 2002). In sum, EMM and indirect inference are clearly attractive methods, given that the sample is large enough to use a rich auxiliary model. Even if this is the case, effort and skill are required to successfully use these methods. In the case of EMM, the documentation of the EMM software package (Gallant and Tauchen, 2004; 2007) makes this clear.

The kernel-based indirect inference (KBII) approach suggested by Billio and Monfort (2003) proposes an entirely nonparametric auxiliary model in place of the EMM's highly parameterized auxiliary model. The use of kernel regression methods is considerably simpler than estimation of models based upon a SNP density with a parametric leading term, since software can be written to use data-dependent rules that tune the fitting process to a given data set with little user intervention. The consistency of the kernel regression estimator ensures a good fit to the data. The main drawback with the KBII estimator is that the binding functions are conditional moments of endogenous variables at certain points in the support of the conditioning variables. How many such points to use, and exactly which points to use require decisions on the part of the econometrician. Billio and Monfort recognize this problem and propose a scoring method to choose the binding functions.

The simulated nonparametric estimators (SNEs) of Altissimo and Mele (2007) are generally applicable, and are asymptotically efficient when the model is Markovian in the observable variables. This is often an important limitation, since models that are Markovian in all variables are usually not Markovian in a subset of the variables (Florens *et al.* 1993). When the model is not Markovian in the observable variables, the proposed SNEs are not asymptotically efficient.

This paper offers a new estimator that is applicable to general DLV models. It is a new implementation of the simulated method of moments (SMM) that allows use of conditional moments. Conditional moments are evaluated using nonparametric kernel smoothing of simulated data. The estimator is very simple to use since it is just an ordinary GMM estimator that uses kernel smoothing to evaluate moment conditions. Because it is a method of moments estimator, it is not in general asymptotically efficient. However, Monte Carlo results show that moment conditions may be chosen such it performs well in comparison to other estimators that have been proposed for estimation of general DLV models. The estimator is referred to as the simulated nonparametric moments (SNM) estimator.

The next section defines the estimator and discusses its properties and usage. The third section presents several examples that compare the SNM estimator to other methods, using

Monte Carlo. Section 4 applies the estimator to weekly spot market exchange rate data, and Section 5 concludes.

2. The SNM estimator

2.1. **Definition of the estimator.** The moment-based estimation framework used in this paper is standard, and is as follows. The sample is $\mathrm{Z}_n = \{(y_t, x_t)\}_{t=1}^{n}$, where y_t is the realization of the k_y dimensional vector of endogenous variables Y_t, and x_t is the realization of the k_x dimensional vector X_t, which is formed of lagged endogenous and exogenous variables. Define the conditional moments $\phi(x_t;\theta) \equiv E\left[Y_t|X_t = x_t;\theta\right]$ (these moments are assumed to exist).

Error functions are of the form

$$\varepsilon(y_t, x_t;\theta) = y_t - \phi(x_t;\theta), \tag{2}$$

An M-estimation approach (Huber, 1964; Gallant, 1987) that down-weights extreme errors will often be used. In this case, error functions are

$$\varepsilon(y_t, x_t;\theta) = \tanh\left(\frac{y_t - \phi(x_t;\theta)}{2}\right) \tag{3}$$

Moment conditions are defined by interacting a vector of instrumental variables $z(x_t)$ with error functions:

$$m(y_t, x_t;\theta) = z(x_t) \otimes \varepsilon(y_t, x_t;\theta) \tag{4}$$

Let the dimension of $z(x_t)$ be k_z. With k_z instruments and $k_y$endogenous variables, the number of moment conditions is $k_y k_z$. Average moment conditions are

$$m_n(\mathrm{Z}_n;\theta) = \frac{1}{n}\sum_{t=1}^{n} m(y_t, x_t;\theta) \tag{5}$$

To simplify the notation, I will often write $m_n(\theta)$ in place of $m_n(\mathrm{Z}_n;\theta)$. The objective function is

$$s_n(\mathrm{Z}_n;\theta) = m_n'(\theta) W(\hat{\tau}_n) m_n(\theta) \tag{6}$$

where $W(\hat{\tau}_n)$ is a weighting matrix that may depend upon prior estimates of nuisance parameters.

Often, $\phi(x_t;\theta)$ in equations 2 and 3 has a known functional form, in which case estimation may proceed using the standard generalized method of moments (GMM). When no closed-form functional form is available it may be possible to define an unbiased simulator $\widetilde{\phi}(x_t, u;\theta)$ such that $E_u\left[\widetilde{\phi}(x_t, u;\theta)\right] = \phi(x_t;\theta)$, where the distribution of u conditional on $X = x_t$ is known. If this is so, a simulated error function can be defined by replacing $\phi(x_t;\theta)$ in equations 2 and 3 with an average of S draws of $\widetilde{\phi}(x_t, u_t^s;\theta)$. Doing so, and then proceeding with normal GMM estimation methods defines the SMM estimator (Gouriéroux and Monfort, 1996, pg. 27). However, in the case of general DLV models, it is often not possible to simulate subject to the conditioning information $X_t = x_t$, as was discussed above.

In this case, the SMM estimator cannot be based upon conditional moments as defined in equations 2-5. Estimation by SMM using unconditional moments is still feasible, but the Monte Carlo evidence cited above has shown that this approach often has poor efficiency, due to the fact that unconditional moments provide little information on the dynamics of a DLV model.

The fundamental idea of the simulated nonparametric moments (SNM) estimator proposed here is to replace the expectations $\phi(x_t;\theta)$ that are used to define error functions in equations 2 and 3 with kernel regression fits based on a very long simulation from the model. Kernel regression (also known as kernel smoothing) is a well-known nonparametric technique for estimating regression functions of unknown form (Robinson, 1983; Bierens, 1987; Härdle, 1991; Li and Racine, 2007). Its application here is entirely standard, except for the use of simulated data.

In the following, tildes will be used to indicate simulated data or elements that depend upon simulated data. Let $\widetilde{Z}_S(\theta) = \{(\widetilde{y}_s(\theta), \widetilde{x}_s(\theta))\}_{s=1}^S$ be a simulated sample of size S from the model, at the parameter value θ. Kernel regression may be used to fit $\phi(x_t;\theta)$, using this simulated data

$$\widetilde{\phi}(x_t;\widetilde{Z}_S(\theta)) = \sum_{s=1}^{S} \widetilde{w}_s \widetilde{y}_s(\theta) \tag{7}$$

where the weight $\widetilde{w}_s$ is

$$\widetilde{w}_s = \frac{K\left(h_S^{-1}\left[x_t - \widetilde{x}_s(\theta)\right]\right)}{\sum_{s=1}^{S} K\left(h_S^{-1}\left[x_t - \widetilde{x}_s(\theta)\right]\right)} \tag{8}$$

To avoid notational clutter, I will often write $\widetilde{\phi}(x_t;\theta)$ in place of $\widetilde{\phi}(x_t;\widetilde{Z}_S(\theta))$ in the following. Note that the same weight $\widetilde{w}_s$ applies to each element of $\widetilde{y}_s$ (which is a k_Y-vector). To speed up computations, one should not separately fit each of the k_Y endogenous variables, but rather employ a specialized kernel fitting algorithm that saves the weights across variables. Since x_t is of dimension k_x, which is in usually greater than one, the kernel function $K(\cdot)$ is in general multivariate. The bandwidth (or window width) parameter is h_S. Note that the kernel regression fit can be evaluated at x_t without requiring that the simulated sequence contain any realizations such that $\widetilde{x}_s = x_t$. What is required for a good fit at x_t is that there there be a large number of realizations that are "close enough" to x_t.

The SNM estimator follows the standard moment-based estimation framework, except that the kernel fit $\widetilde{\phi}(x_t;\theta)$ is used in place of the expectation of unknown form, $\phi(x_t;\theta)$. To be explicit, the SNM estimator is based on error functions of the form

$$\widetilde{\varepsilon}(y_t, x_t; \widetilde{Z}_S(\theta)) = y_t - \widetilde{\phi}(x_t;\theta), \tag{9}$$

or

$$\widetilde{\varepsilon}(y_t, x_t; \widetilde{Z}_S(\theta)) = \tanh\left(\frac{y_t - \widetilde{\phi}(x_t;\theta)}{2}\right) \tag{10}$$

The moment function contribution of an observation is

$$\widetilde{m}(y_t, x_t; \widetilde{Z}_S(\theta)) = z(x_t) \otimes \widetilde{\varepsilon}(y_t, x_t; \widetilde{Z}_S(\theta)) \tag{11}$$

Average moment conditions are

$$\widetilde{m}_n(Z_n; \widetilde{Z}_S(\theta)) = \frac{1}{n}\sum_{t=1}^{n} \widetilde{m}(y_t, x_t; \widetilde{Z}_S(\theta)) \tag{12}$$

To clarify the notation, I will often write $\widetilde{m}_n(\theta)$ in place of $\widetilde{m}_n(Z_n; \widetilde{Z}_S(\theta))$. The objective function that defines the SNM estimator is

$$\widetilde{s}_n(Z_n; \widetilde{Z}_S(\theta)) = \widetilde{m}_n'(\theta) W(\hat{\tau}_n) \widetilde{m}_n(\theta) \tag{13}$$

where $W(\hat{\tau}_n)$ is a weighting matrix that may depend upon prior estimates of nuisance parameters. The SNM estimator is the minimizer of this function:

$$\widetilde{\theta}_n = \arg\min \widetilde{s}_n(Z_n; \widetilde{Z}_S(\theta)). \tag{14}$$

To simplify the notation, the objective functions that define the GMM and SNM estimators will often be written as $s_n(\theta)$ and $\widetilde{s}_n(\theta)$, respectively.

2.2. **Properties of the SNM estimator.** This section deals with the consistency and asymptotic normality of the SNM estimator. The proof offered here is high level, in the sense Assumptions 2 and 3 below are made without detailing assumptions on the DLV model of equations 1 that would cause them to hold. Given a more concrete formulation of the DLV model, one could provide more low level assumptions that would imply Assumptions 2 and 3. This is not done here since the intention is not to focus on any particular model.

The first assumption defines the true parameter value:

Assumption 1. *The sample* $Z_n = \{(y_t, x_t)\}_{t=1}^{n}$ *is generated by the DLV model of equations 1, at the true parameter value* θ_0.

Next, assume that the chosen endogenous variables, conditioning variables, and instruments define a GMM estimator that is consistent and distributed asymptotically normally. Of course, this estimator normally is not feasible if the SNM estimator is under consideration, but abstractly, it is assumed to have the usual desirable properties:

Assumption 2. *Let* $\widehat{\theta}_n = \arg\min s_n(Z_n; \theta)$ *where* $s_n(Z_n; \theta)$ *is defined in equation 6. This (infeasible) GMM estimator is consistent:* $\widehat{\theta}_n \overset{a.s.}{\to} \theta_0$ *and asymptotically normally distributed:* $\sqrt{n}\left(\widehat{\theta}_n - \theta_0\right) \overset{d}{\to} N(0, V_\infty)$ *where* V_∞ *is a finite positive definite matrix.*

Next, assume that the kernel regression estimator used to define the SNM error functions in equations 9 and 10 is strongly consistent, uniformly over the conditioning variables, as the length of the simulation, S, tends to infinite:

Assumption 3. $\widetilde{\phi}_S(x_t; \theta) \overset{a.s.}{\to} \phi(x_t; \theta)$, *for almost all* x_t, *as* $S \to \infty$.

A number of results can justify this assumption, depending on the nature of the model. For example, supposing that the data Z_n generated by the DLV model constitutes a strictly stationary α-mixing sequence, Lu and Cheng (1997) show that Assumption 3 holds.

Assumption 4. *The parameter space Θ over which minimization is done is compact.*

With a compact parameter space, the convergence of Assumption 3 holds uniformly over Θ.

Assumption 5. *The simulation length, S, is greater than the sample size, n.*

This will allow us to focus on asymptotics as n tends to infinity, without separately dealing with S.

Assumption 6. *The instruments are bounded in probability: $z_j(x_t) = O_p(1)$, $j = 1,2,...,k_z$.*

Proposition 1. *This is being worked on*

Proof of Proposition 1: See the Appendix.

By making S suitably large, it is possible to make $\widetilde{\phi}_S(x_t;\theta)$ as close as is desired to the true moment $\phi(x_t;\theta)$. In principle, S could be chosen large enough so that the differences between the error functions in equations 2 and 9 (or the M-estimation analogues in equations 3 and 10) are smaller than the machine precision of a digital computer. If this is the case, the SNM estimator essentially *is* the infeasible GMM estimator.

A simple Monte Carlo exercise illustrates this point. Samples of size $n = 30$ were generated using the classical linear model (CLM)

$$\text{CLM:} \begin{cases} y & = \beta_1 + \beta_2 x + \varepsilon \\ x & \sim U(0,1) \\ \varepsilon & \sim N(0,1) \end{cases} \tag{15}$$

The parameters β_1 and β_2 were randomly drawn (separately) from $U(0,1)$ distributions at each of 1000 Monte Carlo replications. The maximum likelihood (ML) estimator is the ordinary least squares (OLS) estimator obtained by regressing y on a constant and x. The ML estimator may be thought of as a GMM estimator that uses the single ($k_y = 1$) error function $\varepsilon_t = y_t - \beta_1 - \beta_2 x_t$ and the instruments $(1,x_t)$. The SNM estimator was applied, using the endogenous variable y_t, the conditioning variable x_t and instruments $(1,x_t)$. The simulation length was $S = 500000$, and the $h_S = S^{-1/(4+k_x)}$ is chosen using a simple rule-of-thumb procedure [2]. A standard Gaussian kernel was used.

Table 1 gives results that compare the distribution of the difference between the SNM and GMM estimators to the distribution of the GMM estimator, over the 1000 Monte Carlo replications. We can see that the difference between the two estimators is distributed tightly around zero, and that the dispersion of the difference is much less than that of the GMM estimator. If the value of the SNM estimator is regressed on a constant, the value of the

[2] See Li and Racine, 2007, pg. 66. Recall that k_x is the number of conditioning variables ($k_x = 1$ in the present case).

GMM estimator, and the value of the true parameter, the results are (estimated standard errors in parentheses), for the constant, β_1:

$$\widehat{\beta}_1(SNM) = \underset{(0.00023012)}{-0.00106912} + \underset{(0.00030566)}{1.00292}\,\widehat{\beta}_1(GMM) - \underset{(0.00050332)}{0.00267236}\,\beta_1$$

For the slope, β_2, we obtain

$$\widehat{\beta}_2(SNM) = \underset{(0.00038392)}{2.50475\text{e-}5} + \underset{(0.00029626)}{1.00389}\,\widehat{\beta}_2(GMM) - \underset{(0.00073023)}{0.000178451}\,\beta_2$$

In both cases, R^2 is higher than 0.999. We see that the SNM and GMM estimators are essentially identical, independent of the true parameter value.

Recall that the GMM estimator is fully asymptotically efficient for this model. Comparing root mean squared error (RMSE) over the 1000 Monte Carlo replications, the RMSE of the SNM estimator relative to RMSE of the fully efficient GMM estimator is 1.003 in the case of β_1, and 1.004 in the case of β_2. Since the estimators are essentially the same, so are their efficiencies. The SNM estimator can be very efficient if moment conditions are well-chosen.

These results illustrate the fact that when a long enough simulation is used the SNM estimator essentially *is* the GMM estimator that uses the same endogenous variables and the same conditioning variables. The GMM estimator adds information about the functional form of the moment condition, while the SNM estimator fits it nonparametrically. When S is large enough, the nonparametric fit is so good that the SNM estimator is practically identical to the GMM estimator. Of course, one would only use the SNM estimator when the functional form of $\phi(x_t;\theta)$ is unknown, so that the GMM estimator is infeasible.

2.2.1. *Inference and estimation of the optimal weight matrix.* Given that the SNM estimator has the same asymptotic distribution as the infeasible GMM estimator, one can use standard methods and asymptotic results for GMM estimators to make statistical inferences with the SNM estimator. For example, an overidentified model's specification may be tested using the familiar χ^2 test based upon $n\widetilde{s}_n(\widetilde{\theta}_n)$, (assuming that an optimal weight matrix is used).

The asymptotic covariance matrix of the moment conditions is

$$\Omega = \lim_{n\to\infty} \mathscr{E}\left[n\widetilde{m}_n(\theta^0)\widetilde{m}_n(\theta^0)'\right] \tag{16}$$

where $\widetilde{m}_n(\theta^0)$ is defined in equation 12. A consistent estimator of this matrix is needed if one wishes to use an efficient weight matrix, and in any event it is also needed for hypothesis testing. In the ordinary GMM setting without a fully simulable model, this covariance matrix must be estimated using only the sample data, which requires use of one of the kernel-based heteroscedasticity and autocorrelation-consistent covariance matrix estimators (for example, that of Newey and West, 1987). It is well-known that inferences based upon such covariance estimators can be quite unreliable (Hansen, Heaton and Yaron, 1996; Windmeijer, 2005).

In the context of the SNM estimator, or any other moment-based estimator that relies on a fully simulable model, is is possible to estimate Ω though Monte Carlo. The moment conditions of equation 12 may be simulated many (say, R) times, given an initial consistent estimate of the model's parameter. Following notation previously used, a simulation of length equal to the real sample size (n), at the initial consistent estimate $\widetilde{\theta}$ may be represented by $\widetilde{Z}_n(\widetilde{\theta}) = \left\{ \left(\widetilde{y}_t(\widetilde{\theta}), \widetilde{x}_t(\widetilde{\theta}) \right) \right\}_{t=1}^{n}$. We may generate R such samples of size n, and for each of them calculate simulated moment conditions as in equation 12. The r^{th} such replication ($r = 1, 2, ..., R$) is

$$\left(\widetilde{\widetilde{m}}_n\right)_r = \widetilde{m}_n(\left(\widetilde{Z}_n(\widetilde{\theta})\right)_r ; \left(\widetilde{Z}_S(\widetilde{\theta})\right)_r)$$

where $\left(\widetilde{Z}_n(\widetilde{\theta})\right)_r$ and $\left(\widetilde{Z}_S(\widetilde{\theta})\right)_r$ are independent simulations of lengths n and S, respectively. Let $\overline{m}$ be the average of the R draws of $\left(\widetilde{\widetilde{m}}_n\right)_r$, and define $v_r = \left(\widetilde{\widetilde{m}}_n\right)_r - \overline{m}$. Then Ω of equation 16 may be estimated using

$$\widetilde{\Omega} = \frac{n}{R} \sum_{r=1}^{R} v_r v_r' \tag{17}$$

This procedure requires R evaluations of the moment conditions, where R is a reasonably large number. This is not unduly burdensome computationally, since a large number of evaluations of the moment conditions is done during the course of iterative minimization of the objective function $\widetilde{s}_n(\theta)$ of equation 13. If it is computationally feasible to minimize $\widetilde{s}_n(\theta)$, then it is also computationally feasible to estimate Ω using the above procedure. This method has the advantage that it obviates the need for decisions regarding lag lengths, pre-whitening and so forth that attend the use of kernel-based covariance matrix estimators that use only the sample data.

To provide some rudimentary evidence of this covariance estimator's performance, a Monte Carlo study of 1000 replications was done. Data was generated using the classical linear model of equations 15. The SNM estimator was applied using a sample size $n = 30$, a simulated sample size $S = 10000$, and $R = 1000$ draws were used to estimate Ω for each of the 1000 Monte Carlo replications. The true value of Ω for this model is $\Omega_{11} = 1$, $\Omega_{12} = 1/2$, $\Omega_{22} = 1/3$. Over the 1000 Monte Carlo replications, the mean and standard errors (in parentheses) of the replications of $\widetilde{\Omega}$ are Ω_{11} : 1.036 (0.048), Ω_{12} : 0.517 (0.025), Ω_{22} : 0.343 (0.015). For this simple model, the covariance of the moment conditions is estimated quite well using the proposed simulation method. The small upward bias is likely due to the shortness of the simulation length, S. More careful investigation of the empirical performance of this covariance matrix estimator is left for future work.

Once the covariance of the moments is estimated, hypothesis testing may then be done using standard results for GMM estimators with an inefficient weight matrix, or a second round of estimation may be done using the inverse of $\widetilde{\Omega}$ to estimate the efficient weight matrix.

2.2.2. *Choice of the kernel and the bandwidth.* To implement the SNM estimator, the kernel function $K(\cdot)$ in equation 8 must be chosen, as must the bandwidth, h_S. Regarding the kernel, in this paper attention is restricted to local constant kernel regression estimators (Li and Racine, 2007). In this context, much theoretical and empirical evidence shows that the choice of the particular kernel function has relatively little effect on the results, as long as the bandwidth parameter is chosen appropriately, given the kernel (Li and Racine, 2007). For this reason, this paper uses Gaussian product kernels exclusively, accompanied by prior rotation of the data to approximate independence of the conditioning variables. Gaussian product kernels lead to error functions that are continuous and relatively smooth in the parameters, which facilitates iterative minimization. Kernels such as the radial symmetric Epanechnikov are relatively inexpensive to compute, but can lead to error functions that are discontinuous in the parameters, which complicates minimization of the objective function that defines the SNM estimator. This paper leaves the possibility of SNM estimation based on local linear or local polynomial kernel methods for future work.

Given the kernel function, the bandwidth must be chosen. The bandwidth does have an important effect upon the quality of the kernel regression fit. Too large a bandwidth oversmooths the data, and induces a fit with low variance but high bias. Too small a bandwidth has the opposite effect. The bandwidth may be chosen using data-driven methods such as leave-one-out cross validation, or by using rule-of-thumb methods that are known to work well in certain circumstances but may perhaps perform poorly in others. In this paper, a simple rule-of-thumb method is used throughout, since investigation of data-driven methods would add substantially to the computational burden of the Monte Carlo work presented below. It is expected that use of a data-driven method would improve the performance of the SNM estimator. Future work will address this issue more carefully.

2.2.3. *Computational issues.* Estimation of a complicated model using long simulation may become computationally burdensome, since kernel smoothing is a computationally intensive procedure. In common with normal GMM estimators (Chernozhukov and Hong, 2003, especially pp. 296-298), the SNM objective function is not globally convex, so one needs to take care to find the global minimum by using estimation methods such as simulated annealing (Goffe *et al.,* 1994). One may seek to use data-based methods to choose the bandwidth, as well. These factors imply that use of the SNM estimator is computationally intensive. However, kernel regression fitting, which is at the heart of the SNM estimator, is easily parallelized (Racine, 2002; Creel, 2005), as is Monte Carlo work (Creel, 2007). The widespread availability of multicore processors is an invitation to take advantage of parallelization opportunities in econometric work. All of the results reported in this paper were obtained on a computational cluster that provided a total of 16 CPU cores, running the PelicanHPC distribution of GNU/Linux[3]. To give an idea of the computational demands

[3]PelicanHPC is described at http://pareto.uab.es/mcreel/PelicanHPC. It is the evolution of the ParallelKnoppix distribution of GNU/Linux, which was described in Creel (2007).

associated with the SNM estimator, the results reported in this paper required roughly 10 days of computational time on this cluster.

3. Monte Carlo results

This section presents Monte Carlo results that compare the SNM estimator to other estimators that have been proposed for estimation of DLV models. The intention is to show that the SNM estimator can be used to successfully estimate a variety of DLV models, that the SNM estimator performs well in comparison to alternative estimators, and to give examples of how the moment conditions that define the SNM estimator may be chosen.

All of the simulations shared the following features. The SNM estimator was implemented using a Gaussian product kernel. Both the simulated and real conditioning variables were transformed in two ways before applying the Gaussian kernel. First, they were individually shifted and scaled so that their minima and maxima were -4 and 4, respectively. This "compactification" ensures that trial parameter values cannot generate extreme outliers that have no neighbors close enough to generate a positive weight when evaluating the kernel. This transformation is done to provide numeric stability, which is needed when many Monte Carlo replications of a nonlinear minimization are to be done. The second transformation is to multiply by the inverse of the Choleski decomposition of the sample covariance matrix of the real conditioning variables (after the first transformation), before applying the Gaussian kernel. The transformed variables are thus more nearly independent, which makes use of a product kernel more reasonable. In all cases the rule of thumb bandwidth $h = S^{-1/(4+k_x)}$ was used, where k_x is the number of conditioning variables. Likewise, the M-estimation error functions of equation 3 were always used, since they were found to provide good numerical stability during the course of many nonlinear minimizations. Future work could explore efficiency issues with regard to the choice of error functions. In all cases a simulation length of $S = 10000$ was used, to limit the computational burden. For the same reason, only first round estimates using an identity weight matrix were calculated. For each problem, 500 Monte Carlo replications were calculated. Because the SNM objective function is not necessarily globally convex, care is needed to ensure that the global minimum of the objective function is found. For each Monte Carlo replication, minimization was done using an initial course of simulated annealing that involved at least 300 trial values for the parameter vector, followed by use of a quasi-Newton method iterated to convergence.

3.1. Stochastic volatility.

Andersen, Chung and Sorensen (1999) provide Monte Carlo results comparing EMM with GMM in the context of a simple stochastic volatility model. Adapting the notation to conform with the general DLV model of equation 1, the model is

$$\text{SV1:} \begin{cases} y_t & = \exp\left(y_t^*/2\right)\varepsilon_{1t} \\ y_t^* & = \alpha + \beta y_{t-1}^* + \sigma\varepsilon_{2t} \end{cases} \tag{18}$$

where the white noise $\varepsilon_t = (\varepsilon_{1t}, \varepsilon_{2t})'$ is distributed i.i.d. $N(0, I_2)$. The stochastic volatility model of equation 18 will be referred to as SV1. Andersen, Chung and Sorensen apply

GMM using a number of unconditional moments (see Andersen and Sorensen, 1996, for details), and they implement EMM using a number of auxiliary models, including some that use a semi-nonparametric density.

Here I report Monte Carlo results for SNM estimation of this model, using the parameter values $(\alpha,\beta,\sigma) = (-0.736,\ 0.9,\ 0.363)$, which is the case on which Andersen, Chung and Sorensen focus. The sample size is $n = 1000$ observations. The endogenous variables used to define the error functions are y_t^2 and $y_t^2 y_{t-1}^2$(scaled to make their transformed standard errors of the same order of magnitude). The first of these seems a natural choice to provide information on α and σ. The second is intended to capture the temporal correlation of the variance, which should give information on β. The conditioning variable is y_{t-1}^2. The instruments are the same conditioning variable, plus a vector of ones. Two endogenous variables and two instruments imply a total of four moment conditions. Estimation was done by minimizing the objective function in equation 13, using the M-estimation error functions, as in equation 10.

Of the 500 replications, one failed to converge to the specified tolerances for the function, gradient and change in parameters within the limiting number of iterations, though it did not crash. Inclusion or exclusion of this replication does not change the results in any important way. The results presented in Table 2 use the 499 replications that iterated to convergence. These results can be compared to those given in ACS's Table 2 (page 72), which gives results for GMM and EMM estimators, using the same sample size. For purposes of comparison, the last row of Table 2 gives the lowest RMSE from ACS's Table 2. For the α and β parameters, the SNM estimator obtains a considerably lower RMSE than the best of the estimators considered by ACS. In the case of σ, the infeasible GMM estimator and several of the EMM estimators do a little better than the SNM estimator.

Fermanian and Salanié (2004) and Altissimo and Mele (2007) perform Monte Carlo studies using a similar stochastic volatility model, parameterized as

$$\text{SV2:} \begin{cases} y_t & = \sigma_b \exp\left(y_t^*/2\right)\varepsilon_{1t} \\ y_t^* & = \phi y_{t-1}^* + \sigma_\varepsilon \varepsilon_{2t} \end{cases} \tag{19}$$

The stochastic volatility model of equation 19 will be referred to as SV2. The design of the parameters in both of these papers is $(\phi,\sigma_b,\sigma_\varepsilon) = (0.95, 0.025, 0.260)$, and in both cases a sample size of $n = 500$ observations is used. I use the same design and sample size here.

The SNM estimator was used to estimate the SV2 model using the endogenous variables y_t, y_t^2 and $y_t^2 y_{t-1}^2$(with scaling to make the variables' standard errors of the same order of magnitude) and conditioning variables $y_{t-1}$and y_{t-1}^2. The instruments are the same conditioning variable, plus a vector of ones. Three endogenous variables and three instruments imply a total of nine moment conditions used to estimate the three parameters. All of the 500 Monte Carlo replicates converged to the required tolerances. In Table 3 we can see that the SNM estimator gives a considerably more precise estimate of ϕ than do the other estimators. For σ_ε and σ_b, all of the estimators obtain similar RMSEs.

Comparing Tables 2 and 3, we see that the SNM estimator is considerably biased for the σ parameter of the SV1 model and the σ_ε parameter of the SV2 model. Use of different moment conditions or an optimal weight matrix could possible reduce this bias. However, interest usually centers on the autoregressive parameter of the latent process, and for this parameter the SNM estimator performs quite well.

3.2. **Autoregressive Tobit.** Fermanian and Salanié (2004) used an autoregressive Tobit model to illustrate their nonparametric simulated maximum likelihood (NPSML) estimator. This model, with notation adapted to follow the general DLV model of equation 1 of this paper, may be written as:

$$\text{AR Tobit:} \begin{cases} y_t = & \max(0, y_t^*) \\ y_t^* = & \alpha + \beta y_{t-1}^* + \sigma \varepsilon_t \\ \varepsilon_t \sim & IIN(0,1) \end{cases} \tag{20}$$

This model has one observable variable, y_t, a single latent variable, y_t^* and a scalar white noise ε_t. Fermanian and Salanié's Monte Carlo example used the true parameter values $(\alpha, \beta, \sigma) = (0.0,\ 0.5,\ 1.0)$ and the sample size $n = 150$. This same design is used here. To apply the SNM estimator, four error functions are used. The four endogenous variables used to define error functions are y_t (to provide information on α), y_t^2 (to provide information on σ), and $y_t y_{t-1}$ and $y_t y_{t-2}$ (to provide information on β). Each of the four error functions is conditioned on y_{t-1}. The instruments are the same conditioning variable, plus a vector of ones. With 4 endogenous variables and two instruments, a total of 8 moment conditions is used to estimate the three parameters of the model

Table 4 reports the results, along with Fermanian and Salanié's results for comparison. Of the 500 Monte Carlo replications, 499 converged properly. The other replication had not converged within the limiting number of iterations of the quasi-Newton algorithm, and it is dropped (its inclusion does not cause any significant change in the results). The SNM estimator has lower standard errors, but is more biased than the NPSML estimator. For α, the SNM estimator has the lowest RMSE. For β the two estimators have similar RMSEs, and for σ the NPSML estimator has the lowest RMSE. One might note that the conditioning variable in this case is not a strictly continuous random variable, and as such, a Gaussian kernel may not be a good choice. Methods for kernel estimation using mixed discrete/continuous regressors are discussed by Li and Racine (2007).

3.3. **Factor ARCH.** Billio and Monfort (2003) illustrate the kernel-based indirect inference (KBII) estimator with several Monte Carlo examples, one of which is a simple factor ARCH model. The model has a scalar common latent factor, y_t^*, and two observed endogenous variables, $y_t = (y_{1t}, y_{2t})'$. The 2×1 dimensional parameter β has its first element set to

1, for identification. The model, referred to as FA, is

$$\text{FA:}\begin{cases} y_t &= \beta y_t^* + \varepsilon_{1t} \\ y_t^* &= \sqrt{h_t}\varepsilon_{2t} \\ h_t &= \alpha_1 + \alpha_2\left(y_{t-1}^*\right)^2 \end{cases} \tag{21}$$

$t = 1,2,...,n$, where $\varepsilon_{1t} \sim N(0,\sigma^2 I_2)$ and $\varepsilon_{2t} \sim N(0,1)$. The parameter vector design is $(\alpha_1,\alpha_2,\sigma,\beta_2) = (0.2,\ 0.7,\ 0.5,\ -0.5)$.

The error functions for SNM estimation of the FA model were defined using three endogenous variables: the squares of the two components of y_t, and the cross product, $z_t \equiv y_{1t}y_{2t}$. Use of the cross product was found to be helpful for obtaining precise estimates of β_2. These variables were each conditioned on the squares of the two components of y_{t-1} and on the lag of the cross product, z_{t-1}. The instruments were the same conditioning variables, plus a vector of ones. With four instruments and three endogenous variables, a total of 12 moment conditions were used in estimation.

All of the 500 Monte Carlo replications obtained normal convergence. Table 5 reports the results, together with the lowest RMSE that Billio and Monfort obtain using several versions of kernel-based indirect inference, indirect inference, and simulated method of moments (see Billio and Monfort, 2003, Table 5, page 317). For all four parameters, the SNM estimator dominates the estimators considered by Billio and Monfort in terms of lowest RMSE, though for α_2 the bias of the SNM estimator is somewhat larger than one would like.

3.4. **Summary.** This section has illustrated how the SNM estimator may be applied in the estimation of several DLV models. Moment conditions can be chosen with an eye to the information that they provide about specific parameters. The combination of M-estimation error functions, compactification of the conditioning variables, and use of simulated annealing to find good start values lead to a numerically stable estimator that almost always converges. The SNM estimator has been applied subject to several limitations: 1) the simulation length in all cases was quite short ($S = 10000$); 2) the bandwidth parameter (h_S in equation 8) has in all cases been a naive rule-of-thumb rather than a data-based rule that can adapt to the nature of the data that a model generates; and 3) an efficient weighting matrix has not been used.

4. Application: Dollar-Mark exchange rate

To illustrate application of the SNM estimator to real data, and to offer an additional comparison of the SNM estimator to the EMM estimator, this section presents SNM estimates of the parameters of the stochastic volatility model used by Gallant and Tauchen in the User's Guide to the EMM software package (Gallant and Tauchen, 2007) to illustrate the EMM estimator. The data consists of 834 observations of the weekly percentage change of the US dollar to German mark exchange rate, over the years 1975 to 1990. The data is included with the EMM software, and is used here without any alterations. The model used by Gallant and Tauchen, with notation adapted to that of the general DLV model of equation 1, and referred to in the following as the SV3 model, is

$$\text{SV3:} \begin{cases} y_t & = \alpha_0 + \alpha_1(y_{t-1} - \alpha_0) + \exp(y_t^*)\varepsilon_{1t} \\ y_t^* & = \beta_0 + \beta_1(y_t^* - \beta_0) + v_t^* \\ v_t^* & = s\left(r\varepsilon_{1t} + \sqrt{1-r^2}\varepsilon_{2t}\right) \end{cases} \tag{22}$$

This model includes the possibility of correlation between the innovations of the observable and latent variables ("leverage"), though the inclusion of ε_{1t} in the first and third equations. It also allows for slight predictability of returns though the autoregressive term in the first equation.

The SNM estimator is applied using M-estimation error functions, and a simulation length of $S = 100000$. The rule-of-thumb window width is used. The three endogenous variables used to define the error functions are y_t, $z_t \equiv (y_t - \bar{y})^2$ and $z_t z_{t-1}$. The conditioning variables are y_{t-1} and y_{t-2}. The instruments are the same two conditioning variables, plus a vector of ones. There are 3 endogenous variables and three instruments, for a total of 9 moment conditions used to estimate the 6 parameters $\theta = (\alpha_0, \alpha_1, \beta_0, \beta_1, s, r)'$.

Table 6 presents the SNM estimation results, along with EMM estimation results taken from Gallant and Tauchen (2007), for comparison. The first column gives SNM results using an identity weight matrix, while the second column reports results based on the estimated efficient weight matrix (equation 17), using $R = 2000$ replications. Comparing the first and second columns, one may note that use of the efficient weight matrix does not have an important effect on the parameter estimates, nor on the estimated standard errors. The two versions of the SNM estimator give very similar results. Comparing the first two columns with the third, we see that the EMM estimator gives estimates of β_1 and r that are somewhat higher than the SNM estimates. The SNM estimates suggest that leverage is negative ($r < 0$), so that a negative shock to returns is associated with a positive shock to volatility. This is in line with previous evidence (Yu, 2005). The EMM estimates imply a positive leverage effect. For the SNM estimator, the χ^2 test based on the sample size times the objective function value[4] strongly suggests rejection of the model, though the reliability of this test in the case of the SNM estimator is unknown at present. The EMM estimator also suggests that the model might suffer from misspecification, though the p-value of the χ^2 test is not so low as is the case with the SNM estimator.

5. Conclusion

This paper has proposed a simulated method of moments estimator that allows use of conditional moments, in the case of general dynamic latent variable models. The estimator is consistent and asymptotically normally distributed, with the same asymptotic distribution as that of the infeasible GMM estimator defined by the same moment conditions. The Monte Carlo results show that use of conditional moments allows the proposed simulated method

[4]The test statistic is $n\widetilde{s}_n(\widetilde{\theta}_n)$, where $\widetilde{s}_n(\widetilde{\theta}_n)$is given in equation 13. There are 3 degrees of freedom.

of moments estimator to obtain efficiency that is very competitive with other estimation methods.

The SNM estimator relies on the user specifying the moment conditions to use in estimation, as is the case with any method of moments estimator, but the rest of the process can be automatized in software to a high degree. In the present implementation, the kernel function is a Gaussian product kernel, and the bandwidth is chosen using a given rule that depends only on the number of conditioning variables and on the simulation length. One can use the proposed Monte Carlo estimator of the efficient weight matrix that requires no tuning or pre-whitening decisions. Some of the other estimators to which the SNM estimator is compared in this paper require much more active decision making on the part of the modeler. An example is the newer version of the EMM estimator that uses MCMC methods, as presented in Gallant and Tauchen (2007). This version of EMM requires estimation of a SNP density augmented by a leading parametric model to define the score generator. Selection of the parameterization of the score generator is complicated by the fact that it involves many parameters. After estimation of the score generator, the model is estimated using MCMC methods that also require judgement about proper tuning of the Markov chain. Another example is the KBII estimator proposed by Billio and Monfort. Selection of the points at which the binding functions are evaluated is a non-trivial issue which requires judgement. The Monte Carlo results reported here suggest that the SNM estimator can give good performance without requiring the modeler to make any decisions other than the set of moment conditions to use.

The Monte Carlo results provided in this paper show that the SNM estimator achieves root mean squared errors that are often better than those of alternative estimators, and are rarely worse. These results are quite acceptable as they stand, but it is anticipated that they may be improved upon in the future, for two reasons. First, use of an estimated optimal weight matrix is likely to improve efficiency of estimation. Preliminary results suggest that the covariance matrix of the moment conditions can be estimated quite reliably using a Monte Carlo estimator. Future work will investigate the performance of the SNM estimator using an estimated optimal weight matrix. Secondly, a data-based method of choosing the smoothing parameter could improve the fit of the kernel smoother to the true conditional expectations, which would likely improve the results of the SNM estimator. These are simple, obvious extensions to expore. Additional topics for further research include methods to obtain a high precision fit to the conditional moments that define the estimator while using less computational time. Possibilities include the use of sieve estimation methods instead of kernel smoothing, use of approximate nearest neighbors, and use of high performance algorithms for kernel smoothing, such as the improved fast Gauss transform (Yang *et al.*, 2003). Use of an optimal bandwidth may also be helpful for this purpose, since it may be possible to obtain the same quality of fit to $\phi(x_t;\theta)$ while using a shorter simulation length. Another interesting possibility is to attempt to use optimal or approximately optimal instruments. Use of a local linear kernel function instead of the local constant kernel used in this paper would automatically provide estimates of the derivatives $\partial\phi(x_t;\theta)/\partial x_t'$ (see equation

2) of conditional moments with respect to each of the conditioning variables, which could be of use in attempting to approximate optimal instruments.

A CD image that provides the current implementation of the SNM estimator with examples is available on request from the author.

6. APPENDIX

Proof of Proposition 1: As n grows, Assumptions 3 and 5 let us state that

$$\left(\phi(x_t;\theta)-\widetilde{\phi}(x_t;\theta)\right)_j = o_p(1) \tag{23}$$

for $j=1,2,...,k_y$. Recall that there are $k_y k_z$ moment conditions. Considering the difference between $m(x_t;\theta)$ (defined in (4)) and $\widetilde{m}(x_t;\theta)$ (defined in (11)), focus on the q^{th} of the $k_y k_z$ elements of these vectors. Say that this element is the interaction between the j^{th} error function and the r^{th} instrument ($r=1,2,...,k_z$). Then

$$\begin{aligned}
(\widetilde{m}(x_t;\theta)-m(x_t;\theta))_q &= (z(x_t))_r\left(y_t-\widetilde{\phi}(x_t;\theta)\right)_j-(z(x_t))_r\,(y_t-\phi(x_t;\theta))_j \\
&= (z(x_t))_r\left(\phi(x_t;\theta)-\widetilde{\phi}(x_t;\theta)\right)_j \\
&= O_p(1)o_p(1) \\
&= o_p(1),
\end{aligned}$$

by Assumption 6 and equation 23. Averaging over all observations gives

$$\begin{aligned}
(\widetilde{m}_n(\theta)-m_n(\theta))_q &= \frac{1}{n}\sum_{t=1}^{n} o_p(1) \\
&= o_p(n^{-1})
\end{aligned}$$

Furthermore, this holds for all of the moment conditions $q=1,2,...,k_y k_z$. Given this, the objective function that defines the SNM estimator can be written as

$$\begin{aligned}
\widetilde{s}_n(\theta) &= \widetilde{m}_n'(\theta)W(\hat{\tau}_n)\widetilde{m}_n(\theta) \\
&= m_n'(\theta)W(\hat{\tau}_n)m_n(\theta)+o_p(n^{-1}) \\
&= s_n(\theta)+o_p(n^{-1})
\end{aligned}$$

Need to finish this!

TABLES

Table 1: Monte Carlo Results: CLM (equation 15). SNM-GMM is the difference between the SNM and the GMM estimators.

	β_1				β_2			
	Mean	St. Dev.	Min	Max	Mean	St. Dev.	Min	Max
SNM-GMM	-0.001	0.004	-0.015	0.014	0.002	0.007	-0.020	0.025
GMM	0.489	0.473	-0.867	2.143	0.524	0.723	-1.510	2.964

Table 2: Monte Carlo Results: SV1 (equation 18).
* = source: Andersen *et al.* (1999, Table 2, page 72)

	$\alpha = -0.736$			$\beta = 0.90$			$\sigma = 0.363$		
	Mean	St. Dev.	RMSE	Mean	St. Dev.	RMSE	Mean	St. Dev.	RMSE
SNM	-0.653	0.190	0.208	0.916	0.025	0.029	0.285	0.099	0.126
EMM/GMM*	-0.80	0.32	0.33	0.89	0.04	0.04	0.35	0.10	0.10

Table 3: Monte Carlo Results: SV2 (equation 19).
* = source: Fermanian and Salanié (2004, Table 4, page 717)
** = source: Altissimo and Mele (2007, Table 1, page 48)

	$\phi = 0.95$			$\sigma_b = 0.025$			$\sigma_\varepsilon = 0.26$		
	Mean	St. Dev.	RMSE	Mean	St. Dev.	RMSE	Mean	St. Dev.	RMSE
SNM	0.973	0.031	0.038	0.023	0.002	0.003	0.131	0.093	0.158
NPSML*	0.913	0.10	0.107	0.022	0.003	0.004	0.318	0.17	0.180
CD-SNE**	0.909	0.102	0.110	0.024	0.003	0.003	0.229	0.131	0.134
J-SNE**	0.942	0.095	0.095	0.027	0.005	0.005	0.297	0.144	0.149

Table 4: Monte Carlo Results: AR Tobit (equation 20).
* = source: Fermanian and Salanié (2004, Table 1, pg 715)

	$\alpha = 0.0$			$\beta = 0.5$			$\sigma = 1.0$		
	Mean	St. Dev.	RMSE	Mean	St. Dev.	RMSE	Mean	St. Dev.	RMSE
SNM	0.102	0.117	0.155	0.411	0.109	0.140	0.700	0.130	0.327
NPSML*	-0.010	0.215	0.215	0.510	0.151	0.151	0.810	0.184	0.264

Table 5: Monte Carlo Results: FA (equation 21).
* = source: Billio and Monfort (2003, Tables 3, 4 and 5, pp. 313-317). The "other" estimator is that with the lowest RMSE for the given parameter.

	$\alpha_1 = 0.2$			$\alpha_2 = 0.7$		
	Mean	St. Dev.	RMSE	Mean	St. Dev.	RMSE
SNM	0.172	0.087	0.092	0.516	0.134	0.227
Other*	0.244	0.125	0.132	0.659	0.306	0.309
	$\sigma_0 = 0.5$			$\beta_{20} = -0.5$		
	Mean	St. Dev.	RMSE	Mean	St. Dev.	RMSE
SNM	0.503	0.053	0.053	-0.480	0.109	0.111
Other*	0.461	0.135	0.141	-0.445	0.263	0.269

Table 6: Estimation Results: SV3 (equation 22) model for dollar-mark exchange rate
* = source: Gallant and Tauchen (2007, pp. 51-52). The EMM estimate is the mode, and the EMM standard errors are those based upon the Hessian matrix.

	SNM, inefficient weights		SNM, efficient weights $\chi^2(3) = 51.15$ (p=0.000)		EMM* $\chi^2(3) = 8.67$ (p=0.02)	
	Est.	s.e.	Est.	s.e.	Est.	s.e.
α_0	0.048	0.010	0.047	0.010	0.066	0.024
α_1	0.039	0.009	0.038	0.008	0.035	0.028
β_0	-0.048	0.012	-0.044	0.014	0.098	0.098
β_1	0.874	0.007	0.858	0.012	0.940	0.023
s	0.154	0.006	0.165	0.011	0.181	0.019
r	-0.069	0.016	-0.076	0.018	0.117	0.085

REFERENCES

[1] Altissimo, F. and A. Mele (2007) Simulated Nonparametric Estimation of Dynamic Models, working paper.

[2] Andersen, T. and B. Sorensen (1996) GMM estimation of a stochastic volatility model: A Monte Carlo study, *Journal of Business & Economic Statistics*, **14**, 328-352.

[3] Andersen, T., H.-J. Chung and B. Sorensen (1999) Efficient method of moments estimation of a stochastic volatility model: A Monte Carlo study, *Journal of Econometrics,* **91**, 61-87.

[4] Bierens, H. (1987) Kernel estimators of regression functions, in *Advances in Econometrics*, Vol 1 (ed. T. Bewley), Cambridge University Press, 99-144.

[5] Billio, Monica and Alain Monfort (2003) Kernel-based indirect inference, *Journal of Financial Econometrics,* **1**, 297-326.

[6] Carrasco, M., M. Chernov, J.-P. Florens and E. Ghysels (2007) Efficient estimation of general dynamic models with a continuum of moment conditions, *Journal of Econometrics* **140**, 529-573.

[7] Chernozhukov, V. and H. Hong (2003) An MCMC approach to classical estimation, *Journal of Econometrics*, **115**, 293-346.

[8] Creel, M. (2005) User-friendly parallel computations with econometric examples, *Computational Economics*, **26**, 107-128.

[9] Creel, M. (2007) I ran four million probits last night: HPC clustering with ParallelKnoppix, *Journal of Applied Econometrics*, **22**, 215-223.

[10] Fermanian, J.-D. and B. Salanié (2004) A nonparametric simulated maximum likelihood estimation method, *Econometric Theory*, **20**, 701-734.

[11] Fiorentini, Gabriele, Enrique Sentana, and Neil Shephard (2004) Likelihood-based estimation of latent generalized ARCH structures, *Econometrica*, **72**, 1481-1517.

[12] Florens, J.P., M. Mouchart and J. M. Rolin (1993) Noncausality and marginalization of Markov processes, *Econometric Theory*, **9**, 241-262.

[13] Gallant, A.R. (1987) *Nonlinear Statistical Models*, John Wiley & Sons.

[14] Gallant, A. R., and D. Nychka (1987) Seminonparametric maximum likelihood estimation, *Econometrica,* **55**, 363–390.

[15] Gallant, A. R. and G. Tauchen (1996) Which moments to match? *Econometric Theory* **12**, 657– 681.

[16] Gallant, A. Ronald, and George Tauchen (1999) The relative efficiency of method of moments estimators, *Journal of Econometrics*, **92**, 149-172.

[17] Gallant, A. Ronald, and George Tauchen (2002) Simulated score methods and indirect inference for continuous-time models, Chapter in preparation for the *Handbook of Financial Econometrics*.

[18] Gallant, A. Ronald, and George Tauchen (2004) EMM: A program for efficient method of moments estimation: User's Guide Version 1.7, `http://econ.duke.edu/webfiles/get/emm/ver17/guide/`.

[19] Gallant, A. Ronald, and George Tauchen (2007) EMM: A program for efficient method of moments estimation: User's Guide Version 2.5, `http://econ.duke.edu/webfiles/arg/emm/emm.tar`.

[20] Goffe, W.L., G. D. Ferrier and J. Rogers (1994) Global optimization of statistical functions with simulated annealing, *Journal of Econometrics,* **60**, 65-99.

[21] Gouriéroux, C. and A. Monfort (1996) Simulation-based econometric methods, *Oxford University Press*.

[22] Gouriéroux, C., A. Monfort, and E. Renault (1993), Indirect inference, *Journal of Applied Econometrics,* **8**, S85–S118.

[23] Hajivassiliou, V. and D. McFadden (1998) The method of simulated scores for the estimation of limited-dependent variable models, *Econometrica,* **66**, 863–896.

[24] Härdle, W. (1991) *Applied Nonparametric Regression*, Cambridge University Press.

[25] Hansen, L., J. Heaton and A. Yaron (1996) Finite-sample properties of some alternative GMM estimators, *Journal of Business & Economic Statistics*, **14**, 262-280.

[26] Huber, P. (1964) Robust estimation of a location parameter, *Annal of Mathematical Statistics,* **35**, 73-101.

[27] Kristensen, D. and Y. Shin (2006) Estimation of dynamic models with nonparametric simulated maximum likelihood, working paper, http://www.columbia.edu/~dk2313/workingpapers/wp_files/fo-smle-18-01-06.pdf.

[28] Laroque, G. and B. Salanié (1993) Simulation-based estimation of models with lagged latent variables, *Journal of Applied Econometrics,* **8**, S119–S133.

[29] Lee, L.F. (1995) Asymptotic bias in simulated maximum likelihood estimation of discrete choice models, *Econometric Theory,* **11**, 437– 483.

[30] Li, Q. and J. Racine (2007), *Nonparametric econometrics*, Princeton University Press.

[31] Lu, Z. and P. Cheng (1997) Distribution-free strong consistency for nonparametric kernel regression involving nonlinear time series, *Journal of Statistical Planning and Inference,* **65**, 67-86.

[32] McFadden, D. (1989) A method of simulated moments for estimation of discrete response models without numerical integration, *Econometrica,* **57**, 995–1026.

[33] Michaelides, A. and Ng, S. (2000) Estimating the rational expectations model of speculative storage: A Monte Carlo comparison of three simulation estimators, *Journal of Econometrics,* **96**, 231-66.

[34] Newey, W.K. and K.D. West (1987) A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix, *Econometrica*, **55**, 703-08.

[35] Pakes, A. D. Pollard (1989), Simulation and the asymptotics of optimization estimators, *Econometrica,* **57**, 1027–1057.

[36] Racine, Jeff (2002) Parallel distributed kernel estimation, *Computational Statistics & Data Analysis*, **40**, 293-302.

[37] Robinson, P. (1983) Nonparametric estimators for time series, *Journal of Time Series Analysis*, **4**, 185-207.

[38] Smith, A. (1993) Estimating nonlinear time series models using simulated vector autoregressions, *Journal of Applied Econometrics,* **8**, S63–S84.

[39] Tauchen, George (1997) New minimum chi-square methods in empirical finance, in *Advances in Econometrics*, Seventh World Congress, eds. D. Kreps, and K. Wallis, Cambridge UK: Cambridge University Press, 279–317.

[40] Windmeijer, F. (2005) A finite sample correction for the variance of linear efficient two-step GMM estimators, *Journal of Econometrics,* **126**, 25-51.

[41] Yang, C., R. Duraiswami, N. Gumerov and L. Davis (2003) Improved fast Gauss transform and efficient kernel density estimation, Ninth IEEE International Conference on Computer Vision (ICCV'03), **1**, 464-471.

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