Bayesian Estimation of Agent-Based Models

Jakob Grazzini[a,b], Matteo Richiardi[c,d], and Mike Tsionas[e]

[a]Catholic University of Milan, Department of Economics and Finance, Italy.

[b]Catholic University of Milan, Complexity Lab in Economics ,Italy.

[c]Institute for New Economic Thinking and Nuffield College, Oxford, UK.

[d]University of Torino, Department of Economics and Statistics, and Collegio Carlo Alberto, Italy

[e]Lancaster University Management School, UK

November 30, 2015

Abstract

We consider Bayesian inference techniques for Agent-Based (AB) models, as an alternative to simulated minimum distance (SMD). We discuss the specificities of AB models with respect to models with exact aggregation results (as DSGE models), and how this impact estimation. Three computationally heavy steps are involved: (i) simulating the model, (ii) estimating the likelihood and (iii) sampling from the posterior distribution of the parameters. Computational complexity of AB models implies that efficient techniques have to be used with respect to points (ii) and (iii), possibly involving approximations. We first discuss non-parametric (kernel density) estimation of the likelihood, coupled with Markov chain Monte Carlo sampling schemes. We then turn to parametric approximations of the likelihood, which can be derived by observing the distribution of the simulation outcomes around the statistical equilibria, or by assuming a specific form for the distribution of external deviations in the data. Finally, we introduce Approximate Bayesian Computation techniques for likelihood-free estimation. These allow embedding SMD methods in a Bayesian framework, and are particularly suited when robust estimation is needed. These techniques are tested, for the sake of comparison, in the same price discovery model used by Grazzini and Richiardi (2015) to illustrate SMD techniques.

1 Introduction

Agent-based (AB) models are structural dynamical models characterized by three features: (i) there are a multitude of objects that interact with each other and with the environment, (ii) these objects are autonomous, that is there is no central, or 'top-down' control over their behaviour and more generally on the dynamics of the system (e.g. a Walrasian auctioneer), and (iii) aggregation is performed numerically (Richiardi, 2012). Grazzini and Richiardi (2015) show how to apply simulated minimum distance (SMD) techniques to estimate the parameters of AB models, following a frequentist approach. The method of simulated moments (MSM) and indirect inference (II), among other techniques, all fall in this general class. Typically, certain summary statistics have to be selected in advance in order to implement the minimum-distance estimator, requiring additional sensitivity analysis to understand their properties.

In this note, we show how Bayesian methods can be used to perform statistical inference in AB models. The advantages of the Bayesian approach with respect to SMD are threefold: (i) it does not require to pre-select moments (when using the method of simulated moments) or an auxiliary model (in case of indirect inference), (ii) it fully exploits the informational content of the data, hence achieving in theory greater efficiency, (iii) it allows to incorporate prior information, leading to a proper statistical treatment of the uncertainty of our knowledge, and how it is updated given the available observations. The main disadvantage is increased computational costs. To save on these costs, in addition to using efficient sampling schemes in models with large parameters' space, several approximations can be introduced, whose appropriateness should be evaluated on a case-by-case basis. These approximations might also involve giving up (i) and (ii), and resorting again to make inference based on the informational content of (generally insufficient) summary statistics, an appropriate choice of which can also result in more robust estimation.

Bayesian methods are generally used for the estimation of dynamic stochastic general equilibrium (DSGE) models. However, two features of DSGE models make it Bayesian estimation simpler: (i) they produce analytical expressions for the behaviour of the agents around the steady state, (ii) they involve only a limited number of different agents, hence equations (e.g. textbook-version NK models have just three equations). Having analytical expressions for the steady state behaviour allows (log-)linearisation[1] and the application of a simple Kalman filter

[1]Linearisation however is not neutral: it eliminates asymmetries, threshold effects and many other interesting phenomena (Rubio-Ramirez and Fernandez-Villaverde, 2005; Brau et al., 2012).

to derive the likelihood and perform exact inference (on the approximated model). A limited number of equations implies that even if linearisation is not imposed, and a more complicated filter is used, simulation of the model is fast. By contrast, AB models have no closed form equations to describe the behaviour of the agents in the stationary state (if any); moreover, heterogeneity is such that each agent has to be simulated. Consequently, applications of Bayesian techniques has so far been considered out of reach.

The paper is structured as follows. Section 2 formalises AB models; section 3 describes the basic Bayesian techniques that can be used to make inference in AB models, section A introduces our test bed model and presents the results of the different estimation strategies; section 5 offers our concluding remarks.

2 AB models

An Agent-Based (AB) model is a Markov chain where the state of the system at time t is given by the collection of all micro-states at time t $\mathbf{X}_t \equiv \{\mathbf{x}_{it}\}$, $i = 1 \dots N$, $t = 1 \dots T$, with

$$\mathbf{x}_{i,t+1} = \mathbf{f}_i(\mathbf{X}_t, \boldsymbol{\Xi}_t, \boldsymbol{\theta}) \tag{1}$$

where $\mathbf{f}_i$ is a function taking values in $\mathbb{R}^k$, $\boldsymbol{\Xi}_t \equiv \{\xi_{it}\}$ is a vector of stochastic elements and $\boldsymbol{\theta} \in \boldsymbol{\Theta}$ is a parameter vector, with $\boldsymbol{\Theta}$ being a compact subset of $\mathbb{R}^Q$. We assume for simplicity that the model is ergodic, that is, the effects of the random draws $\boldsymbol{\Xi}_t$ fade away with time.

Equation (1) allows us to identify the main differences with respect to DSGE models: the functions $\mathbf{f}$ are typically complicated, possibly involving discontinuities, if-else statements, etc.; even when they are simple, there can be many of them (one for each agent); no equilibrium can be defined in terms of consistency of individual choices.[2]

A set of K aggregate statistics $\boldsymbol{y}_t \equiv \{y_{kt}\}$, $k = 1 \dots K$ can then be defined over $\mathbf{X}_t$:

$$\mathbf{y}_t = \mathbf{m}(\mathbf{X}_t). \tag{2}$$

[2] In rational expectation models, equilibrium is defined as a consistency condition in the behavioural equations: agents (whether representative or not) must act consistently with their expectations, and the actions of all the agents must be mutually consistent. The system is therefore always in equilibrium, even during a phase of adjustment after a shock. By converse, equilibria in AB models are defined only at the aggregate level and only in statistical terms, when macro-observables (eq. 2) become stationary (Grazzini and Richiardi, 2015).

Eqs. (1)-(2) together give

$$\mathbf{y}_{t+1} = \mathbf{g}(\mathbf{X}_t, \boldsymbol{\Xi}_t, \boldsymbol{\theta}) \tag{3}$$

where $\mathbf{X}_t$ is predetermined. If the model is stationary, a long-run statistical equilibrium –called 'absorbing' in Grazzini and Richiardi (2015)– is reached after T^* periods, where the state of the system is independent of the initial conditions $\mathbf{X}_0$ and the seed s which governs the random disturbances $\boldsymbol{\Xi}$:[3]

$$\mathbf{y}^* = E[\mathbf{y}_t | t > T^*] = \mathbf{g}^*(\boldsymbol{\theta}). \tag{4}$$

Typically, we observe data

$$\mathbf{y}^R_{t+1} = \mathbf{g}^R(\mathbf{X}^R_t, \boldsymbol{\Xi}_t, \boldsymbol{\theta}^R, \mathbf{u}_t) \tag{5}$$

and, in equilibrium,

$$\mathbf{y}^R_{t+1} = \mathbf{g}^{*R}(\boldsymbol{\theta}^R, \mathbf{u}_t) \tag{6}$$

where $\mathbf{u}_t$ is a vector of disturbances (accounting for measurement errors, specification errors, etc.). Note that, in general, the micro states $\mathbf{X}_t$ are not observable, and only aggregate data $\bar{\mathbf{X}}_t$ might be available.

SMD techniques work by comparing theoretical constructs computed over $\mathbf{y}_t$, $\boldsymbol{\mu}(\mathbf{y}_t(\boldsymbol{\theta}))$, which depend on the structural parameters $\boldsymbol{\theta}$, with their observed counterparts $\boldsymbol{\mu}^R$, computed on $\mathbf{y}^R$: a value of $\boldsymbol{\theta}$ is selected in order to minimise the distance between the theoretical and observed quantities. Because no closed form expression for the theoretical quantities can be found, they are estimated by simulation, in the artificial data produced by the model. The method of simulated moments (MSM), where the model is summarised ny longitudinal moments, and indirect inference (II), where the model is summarised by the estimated coefficients of an auxiliary model, both fall into this class of estimators, as does simulated maximum likelihood (SML), where the model is summarised by the probability of reproducing the raw observed data. With respect to SML, standard Bayesian methods use the likelihood to derive a posterior distribution of the parameters, as described in the next section.

[3]Non-ergodicity implies the existence of multiple equilibria. In a Bayesian framework, if a model is non-ergodic the simulated data should come from different replications of the model with different random seeds, rather than from one (longer) simulation run. By converse, if the model is ergodic fixing the random seed allows a more precise estimation, as the changes in model behaviour when the parameters are changed are smoother.

Note that $\mathbf{y}$ can be any transformation of the data. As an extreme example, in chaotic systems they may be properties of the orbits / attractors.

3 Methods

The fundamental equation for Bayesian methods is a simple application of the Bayes theorem:

$$p\left(\boldsymbol{\theta}|\mathbf{Y}^R\right) \propto \mathscr{L}\left(\boldsymbol{\theta};\mathbf{Y}^R\right) p\left(\boldsymbol{\theta}\right) \tag{7}$$

where $p\left(\boldsymbol{\theta}\right)$ is the prior distribution of the parameters, $\mathscr{L}\left(\boldsymbol{\theta};\mathbf{Y}^R\right) \equiv p\left(\mathbf{Y}^R|\boldsymbol{\theta}\right)$ is the likelihood of observing the data $\mathbf{Y}^R \equiv \left\{\mathbf{y}_t^R\right\}$, $t = 1 \dots T$ given the value of the parameters, and $p\left(\boldsymbol{\theta}|\mathbf{Y}^R\right)$ is the posterior distribution, that is the updated distribution once the information coming from the observed data is properly considered. The prior distribution typically comes from other studies or subjective evaluations. A uniform distribution in the allowed range of the parameters is often used as a way to introduce 'uninformative' priors, though not such a thing as an uninformative prior actually exists (Bernardo, 1997).[4] What matters, the prior is a distribution, which through application of Bayes theorem produces another distribution as an output (by converse, maximising the likelihood would only produce a point estimate).

Sampling the posterior distribution $p\left(\boldsymbol{\theta}|\mathbf{Y}^R\right)$ involves two computationally intensive steps: (i), for given values of $\boldsymbol{\theta}$, obtaining the likelihood $\mathscr{L}$, (ii) iterating over different values of $\boldsymbol{\theta}$. The resulting posterior is then normalized to have unit integral via a Simpson's rule, a common method for numerical integration.

Sections 3.2 to 3.3 below deal with problem (i), while section 3.4 is devoted to likelihood-free emethods and section 3.5 discusses problem (ii).

3.1 Non-parametric density estimation

Computation of the likelihood, for any given value of $\boldsymbol{\theta}$, is conceptually straightforward. Assuming we are in a statistical equilibrium, the probability of observing the whole (unordered) series of data $\mathbf{Y}^R \equiv \left\{\mathbf{y}_t^R\right\}$ is simply

$$\mathscr{L}\left(\boldsymbol{\theta};\mathbf{Y}^R\right) \propto \prod_{t=1}^{T} f\left(\mathbf{y}_t^R|\boldsymbol{\theta}\right). \tag{8}$$

where we assume to be 'blind' with respect to all the other data points $\left\{\mathbf{y}_{-t}^R\right\}$ when we evaluate $\mathbf{y}_t^R$. Any autocorrelation in $\{\mathbf{y}_t\}$ would then lead to an increase in the variance of the estimated

[4] Sometimes a maximum entropy distribution is used as a 'minimally informative' prior; this however requires (a) some level of information on some moments of the prior to specify the constraints and (b) the choice of a reference measure in continuous settings.

distribution of $\mathbf{y}_t$, and in turn to an increase in the variance of $\mathscr{L}\left(\boldsymbol{\theta}; \mathbf{Y}^R\right)$.[5]

All we need is an estimate for the distribution $\tilde{f}(\boldsymbol{\theta})$; we then evaluate the estimated distribution $\tilde{f}(\boldsymbol{\theta})$ at each observed $\mathbf{y}_t^R$, and compute $\prod \tilde{f}(\mathbf{y}_t^R|\boldsymbol{\theta})$.

Estimation of the density, for any value of $\boldsymbol{\theta}$, is done by simulation: in a statistical equilibrium, the outcome fluctuates around a stationary level $\boldsymbol{y}^*(\boldsymbol{\theta}) = E[\mathbf{y}_t(\boldsymbol{\theta})|t > \bar{T}]$. If we collect the artificial data produced by the model in such a statistical equilibrium, we can construct a probability distribution around $\boldsymbol{y}^*$, and therefore evaluate the density at each observed data point $\boldsymbol{y}_t^R$. If the outcomes $\mathbf{y}(\boldsymbol{\theta})$ were discrete, we would only have to count the frequency of occurrence of each observed value $\mathbf{y}_t^R$. With continuous $\mathbf{y}(\boldsymbol{\theta})$, the likelihood has to be estimated either non-parametrically or parametrically, under appropriate distributional assumptions. A traditional *non-parametric* method is kernel density estimation (KDE), which basically produces histogram-smoothing: the artificial data are grouped in bins (the histogram), and then a weighted moving average of the frequency of each bin is computed.[6] The approximation bias introduced by KDE can be reduced by using a large number of very small bins, but then the variance in the estimate of the density grows. To see this, think about estimating the probability density function (PDF) when the data comes from any standard distribution, like an exponential or a Gaussian. We can approximate the true PDF $f(x)$ to arbitrary accuracy by a piecewise-constant density (that is, by constructing an histogram), but, for a fixed set of bins, we can only come so close to the true, continuous density.

3.2 Parametric estimation of the likelihood

The main problem with KDE is its computational cost. A (faster) alternative is to assume a *parametric* distribution for the density around $\mathbf{y}^*(\boldsymbol{\theta})$:

$$\mathbf{y}_{t+1} = \mathbf{g}^*(\boldsymbol{\theta}) + \boldsymbol{\epsilon}_t. \tag{10}$$

[5] As an example, think of an AR(1) process with mean 0: $y_{t+1} = \rho y_t + \varepsilon_t$, where ρ is the autocorrelation coefficient and $\varepsilon_t \sim \mathscr{N}(0, \sigma_\varepsilon^2)$. The unconditional distribution of y_t is $y_t \sim \mathscr{N}\left(0, \frac{,\sigma_\varepsilon^2}{1-\rho^2}\right)$: an increase in the autocorrelation coefficient ρ increases the variance of the distribution.

[6] More formally, kernel density estimation (KDE), given a simulated time series $\mathbf{y}(\boldsymbol{\theta}) \equiv \{\mathbf{y}_s\}$, $s = 1 \dots S$, approximates the density $f(\mathbf{y}_t^R, \boldsymbol{\theta})$, for each observed data point $\mathbf{y}_t^R$, with:

$$\tilde{f}(\mathbf{y}_t^R|\boldsymbol{\theta}) = \frac{1}{Sh} \sum_{s=1}^{S} \mathscr{K} \frac{\sqrt{\sum_{k=1}^{K} \sum_{s=1}^{S} \left(y_{ks}^S - y_{kt}^R\right)^2}}{h} \tag{9}$$

where $\mathscr{K}$ is a kernel function that places greater weight on values y_{ks} that are closer to y_{kt}^R, is symmetric around zero and integrates to one, and h is the bandwidth. Algorithms for KDE are available in most statistical packages.

Imposing additional information about the distribution of the simulation output can help generate better estimates from a limited number of simulation runs. On the other hand, those estimates may increase the bias if the assumed distribution does not conform to the true density of the model output. Such an assumption can of course be tested in the artificial data, in the relevant range of $\boldsymbol{\theta}$ (i.e. where the estimated coefficients lie). Use of parametric methods leads to *syntetic likelihood* or *pseudo-likelihood* estimation (Wood, 2010; Hartig et al., 2011).

The parametric and non-parametric methods discussed above use the output variability that is predicted by the model, and use this information for inference. However, it is possible that the model shows much less variability than the data. This can be due to fundamental *specification errors* (the model is only a poor approximation of the real process, so that the mean model predictions do not fit to the data), *simplification errors* (the model is a good approximation of the real process, but there are additional stochastic processes that have acted on the data and are not included in the model) or *measurement errors* (there is uncertainty about the data). While the first type of errors calls for a re-specification of the model, the latter types could in principle be dealt with by including additional processes that explain this variability within the model. "However, particularly, when those processes are not of interest for the scientific question asked, it is simpler and more parsimonious to express this unexplained variability outside the stochastic simulation. One way to do this is adding an *external error model* with a tractable likelihood on top of the results of the stochastic simulation" (Hartig et al., 2011):

$$\mathbf{y}_{t+1}^{R} = \mathbf{g}^{*}(\boldsymbol{\theta}^{R}) + \boldsymbol{u}_{t}. \tag{11}$$

The validity of such a strategy depends on the quality of the assumption about the distribution of these external errors, given the model and the data. This assumption cannot be tested *per se*, as the variability in the real data comes both from the *explained* components (i.e. the model) and from the *unexplained* ones (the external errors), the external errors being defined as a residual.

The two parametric strategies (modelling the variability of model outcome and modelling external errors that might affect the real data, in the stationary state) are often explored separately. For instance, most studies that use the augmentation by external errors approach then treat the model outcome as deterministic, in the stationary state, whilst most studies that employ a synthetic likelihood do not consider external errors.[7]

[7]It is in principle possible to combine the two approaches together, and explicit distributional assumptions

Under the assumption that the distribution for $\boldsymbol{\epsilon}$ or $\mathbf{u}$ is Gaussian, the likelihood is also Gaussian, with a mean which depends on $\boldsymbol{\theta}$. We are then able to write down an expression for the likelihood –hence the posterior distribution, under a convenient specification of the priors (see the Appendix). However, the likelihood needs in any case to be simulated, as $\mathbf{g}^*(\boldsymbol{\theta})$ has no closed form expression. Therefore, the assumption of normality is easy to be replaced, if other distributions appear to fit the data better than the normal, or there are theoretical reasons to believe that the external errors are non-Gaussian.

3.3 Unconditional vs. conditional estimation

An important difference between estimation of AB models and estimation of models with exact aggregation results (as in DSGE models) is that in the latter case the evolution of the aggregate state of the system, $\boldsymbol{y}_{t+1}$, depends only on the current macro-state $\mathbf{y}_t$ and the parameters $\boldsymbol{\theta}$, with the set of micro states $\mathbf{X}_t$ not providing any further information. In terms of eq. (3), we have

$$\mathbf{y}_{t+1} = \mathbf{g}(\mathbf{y}_t, \boldsymbol{\xi}_t, \boldsymbol{\theta}) \tag{3'}$$

This can be restated in more general terms by saying that the aggregate representation of the system, which is a projection of the original Markov chain where the state of the system at time t is given by the collection of all the micro-states, is still a Markov chain –a condition called *lumpability* (Kemeny and Snell, 1976).[8] In such a case a better estimation of the likelihood can be obtained.

Let $f\left(\mathbf{y}_{t+1}|\mathbf{y}_t, \boldsymbol{\theta}\right)$ denote the distribution of $\mathbf{y}$ at time $t+1$ given the observables at t; the likelihood function can be expressed as

$$\mathscr{L}\left(\boldsymbol{\theta}; \mathbf{Y}^R\right) = f\left(\mathbf{y}_0^R|\boldsymbol{\theta}\right) \prod_{t=0}^{T-1} f\left(\mathbf{y}_{t+1}^R|\mathbf{y}_t^R, \boldsymbol{\theta}\right). \tag{8'}$$

Assuming $f\left(\mathbf{y}_0^R|\theta\right) \propto$ const. or that the initial state of the system is known with certainty, it is possible to initialise at any time step t the simulations with the observed values $\mathbf{y}_t^R$, run D different one-period ahead simulations (rather than 1 'long' simulation lasting D periods) and estimate the conditional densities using $\tilde{f}(\mathbf{y}_{t+1}^R|\mathbf{y}_t^R, \boldsymbol{\theta})$ instead of $\tilde{f}(\mathbf{y}_t^R|\boldsymbol{\theta})$ in (8). The scheme is more efficient because the simulated time series are constrained to remain closer to the observed

for both the model outcomes and the external errors, though there are no practical advantages.

[8] There are necessary and sufficient conditions for lumpability, and they have to do with symmetries in the micro-state space (Banish et al., 2012).

ones, hence the likelihood is estimated with more precision.[9] It also allows to relax the weakly stationarity assumption, as conditioning on $\mathbf{y}_t^R$ controls for persistence in the process.[10]

Note that eq. (8′) holds generally. However, when the system is non lumpable the precise way all the individual states $\mathbf{X}_t$ are re-initialised, in order to simulate the system, does matter. Moreover, there might be multiple combinations of micro states $\mathbf{X}_t$ which give the same macro state $\mathbf{y}_t^R$, but different evolution $\mathbf{y}_{t+1}$.[11] In general, if eq. 3′ does not hold (and in AB models it typically does not hold, otherwise it would be more convenient to use a representative agent formulation), and the macro-state $\boldsymbol{y}_{t+1}$ depends on the *distribution* of the unobserved micro-states $\mathbf{X}_t$, the unconditional approach is the only one feasible.

3.4 Likelihood-free methods

As we have seen, obtaining a non-parametric estimate of the likelihood can be computationally heavy. Turning to parametric estimates, under the assumption of a fixed distributional form of the variable of interest around a long-term stationary state predicted by the model —where the variability is produced either by model uncertainty or external errors— can sometimes be too restrictive. Originating from population genetics (Tavaré et al., 1997; Fu and Li, 1997), where the task of estimating the likelihood of the observed changes in DNA is impervious, a new set of methods have appeared in the last fifteen years to produce approximations of the posterior distributions without relying on the likelihood. These methods are labelled 'likelihood-free' methods, and the best known class is approximate Bayesian computation (ABC).[12]

In standard Bayesian methods, it is the likelihood function that provides the fit of the model with the data —describing how plausible a particular parameter set θ is. The likelihood is however often computationally impractical to evaluate. The basic idea of ABC is to replace the evaluation of the likelihood with a 0-1 indicator, describing whether the model outcome is close enough to the observed data. To allow such an assessment, the model outcome and the data must first be summarised. Then, a distance between the simulated and the real data is computed. The model is assumed to be close enough to the data if the distance falls within the admitted tolerance. As such, there are three key ingredients in ABC: (i) the selection of *summary statistics*, (ii) the definition of a *distance measure*, (iii) the definition of a *tolerance threshold*.

[9] To continue with our AR(1) example, the conditional distribution of y_{t+1} given y_t and ρ is $y_{t+1} \sim \mathcal{N}(\rho y_t, \sigma_\varepsilon^2)$.

[10] If persistence is of order higher than 1, conditioning on periods previous to t is required.

[11] The latter would not be a problem if only we could be sure to draw the micro states $\mathbf{X}_t$ randomly from the set of micro states which aggregate to $\mathbf{y}_t^R$, but in practice we can never be sure that this is the case.

[12] See Marin et al. (2011); Turner and Zandt (2012).

The choice of a distance measure is usually the least controversial point (the Euclidean distance or weighted Euclidean distance, where the weights are given by the inverse of the standard deviation of each summary statistics, is generally used). The choice of a tolerance threshold, as we shall see, determines the trade-off between *sampling error* and *approximation error*, given computing time. The choice of summary statistics is the most challenging, and we will discuss it in greater details.

The basic ABC algorithm works as follows:

1. a candidate vector $\boldsymbol{\theta}^c$ is drawn from a prior distribution;

2. a simulation is run with parameters vector $\boldsymbol{\theta}^c$, obtaining simulated data from the model density $p(\boldsymbol{y}|\boldsymbol{\theta}^c)$;

3. the candidate vector is either retained or dismissed depending whether the distance between the summary statistics computed on the artificial data $\boldsymbol{\mu}(\boldsymbol{y}(\boldsymbol{\theta}))$ and summary statistics computed on the real data $\boldsymbol{\mu}(\boldsymbol{y}^R)$ is within or outside the admitted tolerance h: $d(\boldsymbol{\mu}, \boldsymbol{\mu}^R) \leq h$.

This is repeated N times; the retained values of the parameters define an empirical approximated posterior distribution. KDE can then be applied to smooth out the resulting histogram, and obtain an estimate of the theoretical approximated posterior. Approximation error refers to the fact that the posterior is approximated; sampling error refers to the fact that we learn about the approximated posterior from a limited set of data.

It is easy to see where the approximation error comes from. While the true posterior distribution is $p(\boldsymbol{\theta}|\mathbf{y} = \mathbf{y}^R)$, in ABC we get $p(\boldsymbol{\theta}|\boldsymbol{\mu}(\mathbf{y}) \approx \boldsymbol{\mu}(\mathbf{y}^R))$.

If we set the tolerance threshold $h = 0$, and our statistics were *sufficient summary statistics*[13], we would get back to standard Bayesian inference, and sample from the exact posterior distribution. However —and this is the whole point— because of the complexity of the underlying model the likelihood of observing the real data is tiny everywhere, so that acceptances are impossible, or at least very rare. When h is too small, the distribution of accepted values of $\boldsymbol{\theta}$ is closer to the true posterior, and the approximation error is smaller; however, the number of acceptances is usually too small to obtain a precise estimate of the (approximated) posterior distribution: the sampling error increases. On the other hand, when h is too large, the precision

[13] A summary statistics is said to be sufficient if "no other statistic that can be calculated from the same sample provides any additional information as to the value of the parameter" (Fisher, 1922). Sufficient statistics satisfy $p(\boldsymbol{\theta}|\boldsymbol{\mu}(\mathbf{y}^R)) = p(\boldsymbol{\theta}|\mathbf{y}^R)$.

of the estimate improves because we have more accepted values (the sampling error goes down), but the approximation error gets bigger. In other words, we obtain a better estimate of a worse object.

An alternative to choosing h in advance is to specify the number of acceptances k required (e.g. $k = 500$): then, h is chosen (after the distance for every draw is computed) in order to achieve that number of acceptances. Finally, note that the trade-off between sampling error and approximation error is for a given number of draws (hence, a given computing time). Drawing more candidates allows to reduce the approximation error (by decreasing h) without increasing the sampling error. Stated more formally, ABC converges to true posterior as $h \to 0, N \to \infty$.

The choice of summary statistics is at the same time the weak point of ABC and a great source of flexibility. For instance, by choosing as summary statistics longitudinal moments, or the coefficients of an appropriate auxiliary model, it allows to embed the method of simulated moments and indirect inference in a Bayesian setting, incorporating prior information. Also, an appropriate choice of the summary statistics allows to make *conditional forecasts* about the evolution of the real world. Suppose an extreme case where we only condition on the state of the system at time t: we wish to project the likely evolution of a system given $\mathbf{y}_t$. We can then simply set our summary statistics $\boldsymbol{\mu}(\mathbf{y}) = \mathbf{y}_t^R$: the ABC algorithm will retain any simulated trajectory that passes for $\mathbf{y}_t^R$, producing not only a (quite poor, in this case) approximation of the posterior, but also conditional projections about future states.[14]

Any condition can in principle be used as a summary statistics: of course, the lower the informational content of the condition, the poorer the approximation (and the bigger the dispersion of the projections). However, there is also a drawback in increasing the informational content of the summary statistics, and it comes again from the trade-off between sampling error and approximation error. As Beaumont et al. (2002, p. 2026) put it, "A crucial limitation of the [...] method is that only a small number of summary statistics can usually be handled. Otherwise, either acceptance rates become prohibitively low or the tolerance [...] must be increased, which can distort the approximation." This is because the asymptotic rate of converge of ABC to the true posterior distribution, as $h \to 0, N \to \infty$, worsens with $\dim(\boldsymbol{\mu})$. The problem of choosing appropriate *low dimensional* summary statistics that are informative about $\boldsymbol{\theta}$ is an

[14]The case when it is possible to kill two birds (inference and conditional statements) with one stone is of course quite a lucky one. More in general, when the condition in the conditional statement is too poor to allow for good inference, we should keep the two problems separate: *first*, get a good approximation of the posterior (by selecting appropriate summary statistics); *then*, sample from the estimated posterior and select the trajectories that fulfil the condition.

open issue in ABC. "The insidious issue is that it is rarely possible to verify either sufficiency or insufficiency. Furthermore, if they are insufficient, it is usually not possible to determine how badly they have distorted results. Said another way, you know you are probably making errors, but you don't know how large they are" (Holmes, 2015).

The topic is an active area of research. Recent years have seen the development of techniques that provide guidance in the selection of the summary statistics (see e.g. Fearnhead and Prangle, 2012). Also, post-processing of the results can improve the quality of the approximation by correcting the distribution of $\boldsymbol{\theta}$ by the difference between the observed and simulated summary statistics (Beaumont et al., 2002). Efficiency can be improved by assigning a *continuation probability* to each simulation: the idea is to stop prematurely simulations that are likely to end up in a rejection, and has originated the *lazy ABC* approach (Prangle, 2014). Finally, new methods have appeared that require no summary statistics, external error terms, or tolerance thresholds, at a computational cost (Turner and Sederberg, 2013).

3.5 Sampling from the posterior distribution

Application of the Bayes theorem, once the likelihood is known, allows to get a density for the posterior distribution, at *one* given value of $\boldsymbol{\theta}$. However, to recover the whole shape of the posterior distribution, many values have to be sampled. In simple models, exploration of the parameters space can be accomplished by 'brute force' grid exploration: the parameters space is sampled at regular (small) intervals. For instance, if there are two parameters that can potentially vary continuously between 0 and 1, and we set the value of the step to .1, we have 11 values to consider for each parameter, and their combination gives 121 points to sample: by discretising the parameters, we have reduced the size of the parameters space from $\mathbb{R}^2$ to 121 points. Multi-level grid search, where the grid is explored at smaller intervals in ranges of the parameters' space on the bases of the results of previous, looser, grid explorations, can be devised to improve efficiency. However, the curse of dimensionality –the fact that when the dimensionality increases, the volume of the parameters' space increases so fast that sampling becomes sparse– precludes adopting such an approach except when the number of the parameters is small. Grid exploration involves evaluating the density of the posterior distribution at many points where it is practically zero, while more likely values of $\boldsymbol{\theta}$, where a finer search might be valuable, are sampled with the same probability. Efficient sampling involves devising algorithms where the sampling probability, rather than being constant, is proportional to the

posterior density.

There are four main classes of *efficient sampling schemes*, to obtain samples from a function of $\boldsymbol{\theta}$, the *target distribution* (the posterior, in our case), which is unknown analytically but can be evaluated point-wise for each $\boldsymbol{\theta}$: (i) rejection sampling, (ii) importance sampling, (iii) Sequential Monte Carlo, and (iv) Markov chain Monte Carlo,. Here we provide only an intuition of how they work, drawing extensively from the excellent survey by Hartig et al. (2011).[15]

Rejection sampling (RS).* The simplest possibility of generating a distribution that approximates $\mathscr{L}(\boldsymbol{\theta})$ is to sample random parameters $\boldsymbol{\theta}$ and accept those proportionally to their (point-wise approximated) value of $\mathscr{L}(\boldsymbol{\theta})$. This approach can be slightly improved by importance sampling or stratified sampling methods such as the Latin hypercube design, but rejection approaches encounter computational limitations when the dimensionality of the parameter space becomes larger than typically 10-15 parameters.

Importance sampling (IS). The intuition behind importance sampling is to to study the distribution $\mathscr{L}(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\mathbf{y})$ while sampling from another, simpler distribution $q(\boldsymbol{\theta})$ (called *importance distribution*). This technique was born as a variance reduction technique, aimed at increasing the likelihood to sample from an 'important' but small region by sampling from a different distribution that overweights the important region (hence the name). Having oversampled the important region, we have to adjust our estimate somehow to account for having sampled from this other distribution. This is done by re-weighting the sampled values by the adjustment factor $p(\boldsymbol{\theta})/(q(\boldsymbol{\theta}))$. Importance sampling and rejection sampling are similar in as much both distort a sample from one distribution in order to sample from another. They also share the limitation that they do not work well in high dimensions.

Sequential Monte Carlo methods (SMC).** Particle filters or sequential Monte Carlo methods (SMCs) work by filtering proposed values for $\boldsymbol{\theta}$ to arrive at a sample of values drawn from the desired distribution. In SMC each step of the algorithm contains N parameter combinations $\boldsymbol{\theta}_i$ (particles), that are assigned weights ω_i proportional to their likelihood or posterior value $\mathscr{L}(\boldsymbol{\theta}_i)$ (see Arulampalam et al., 2002). When starting with a random sample of parameters, many particles may be assigned close to zero weights, meaning that they carry little information for the inference (degeneracy). To avoid this, a resampling step is usually added where a new set of particles is created based on the current weight distribution [...]. The tra-

[15]The entries marked with a '*' are excerpt from Hartig et al. (2011), where we have replaced ϕ in their notation with $\boldsymbol{\theta}$, in order to maintain consistency. The entries marked with a '**' are based on Hartig et al. (2011), appropriately integrated.

ditional motivation for a particle filter is to include new data in each filter step, but the filter may also be used to work on a fixed dataset or to subsequently add independent subsets of the data.

The advantage of SMC –and of Markov chain Monte Carlo, see below– is that the time needed to obtain acceptable convergence is typically much shorter than for rejection sampling, because the sampling effort is concentrated in the areas of high likelihood or posterior density.

Markov chain Monte Carlo (MCMC).** MCMC sampling also try to concentrate the sampling effort in the areas of high likelihood or posterior density, based on previous samples. MCMC algorithms construct a Markov chain of parameter values $(\boldsymbol{\theta}_1 \dots \boldsymbol{\theta}_n)$, where the next parameter combination $\boldsymbol{\theta}_{i+1}$ is chosen by proposing a random move conditional on the last parameter combination $\boldsymbol{\theta}_i$, and accepted conditional on the ratio of $\mathscr{L}(\boldsymbol{\theta}_{i+1})/\mathscr{L}(\boldsymbol{\theta}_i)$. Given that certain conditions are met (see, e.g. Andrieu et al., 2003), the Markov chain of parameter values will eventually converge to the target distribution $\mathscr{L}(\boldsymbol{\theta})$.

There are a number of MCMC samplers, the most popular of which is the MetropolisHastings algorithm. In its simplest form, the *random-walk Metropolis-Hastings*, in each period a candidate $\boldsymbol{\theta}^c \sim \mathscr{N}\left(\boldsymbol{\theta}^{(s)}, \mathbf{V}\right)$ is drawn, given the current value $\boldsymbol{\theta}^{(s)}$. The candidate is accepted with probability

$$\alpha = \min\left\{1, \frac{p\left(\boldsymbol{\theta}^c|\mathbf{y}_R\right)}{p\left(\boldsymbol{\theta}^{(s)}|\mathbf{y}_R\right)}\right\} \tag{12}$$

in which case we set $\boldsymbol{\theta}^{(s+1)} = \boldsymbol{\theta}^c$; else, we set $\boldsymbol{\theta}^{(s+1)} = \boldsymbol{\theta}^{(s)}$ and we repeat the previous candidate.

With multiple parameters, a slightly more sophisticated version of the basic Metropolis-Hastings algorithm can be used, which resembles the Gibbs sampler and is most useful when choosing a particular covariance matrix $\mathbf{V}$ is not easy. The Gibbs sampler relies on drawing a sample $\left\{\boldsymbol{\theta}^{(s)}, s = 1 \dots S\right\}$ by drawing repeatedly from the conditional distributions

$$\theta_j|\mathbf{Y}^R, \boldsymbol{\theta}_{-j}, j = 1 \dots k$$

When these posterior conditional distributions are not known families, we use the following. Draw a candidate θ_j^c uniformly from the interval (a_j, b_j). The candidate is accepted with probability

$$\alpha = \min\left\{1, \frac{p\left(\theta_j^c|\mathbf{Y}^R, \boldsymbol{\theta}_{-j}^{(s)}\right)}{p\left(\theta_j^{(s)}|\mathbf{Y}^R, \boldsymbol{\theta}_{-j}^{(s)}\right)}\right\} \tag{13}$$

in which case we set $\theta_j^{(s+1)} = \theta_j^c$, otherwise we repeat the previous draw, $\theta_j^{(s+1)} = \theta_j^{(s)}$. The interval (a_j, b_j) is adjusted automatically during the transient or burn-in phase of the algorithm so that approximately 25% of the candidates are accepted.

A final note concerns efficient sampling in an ABC setting. The standard scheme for ABC is, as we have seen, rejection sampling. Candidates are drawn from the prior distribution, and only those that 'perform well' are retained. This is not very efficient, especially if the prior distribution differs significantly from the posterior. However, it is possible to employ ABC with more efficient sampling schemes (see Sisson et al., 2016). For instance, rather than sampling from the prior one could sample from an importance distribution $q(\boldsymbol{\theta})$. Candidate are then accepted if $d(\boldsymbol{\mu}(\boldsymbol{\theta}), \boldsymbol{\mu}^R) \leq h$, with a weight $p(\boldsymbol{\theta})q(\boldsymbol{\theta})$. SMC methods can then be employed to adaptively refine both the threshold and the importance distribution. MCMC methods can also be employed, where new candidates depend on the current value of $\boldsymbol{\theta}$ and are accepted with a modified Metropolis-Hastings rule:

$$\alpha = \min \left\{ 1, \frac{p(\boldsymbol{\theta}^c) \mathbb{1}[d(\boldsymbol{S}(\boldsymbol{\theta}^c), \boldsymbol{S}^R) \leq h]}{p(\boldsymbol{\theta}^{(s)}) \mathbb{1}[d(\boldsymbol{S}(\boldsymbol{\theta}^{(s)}), \boldsymbol{S}^R) \leq h]} \right\} \tag{14}$$

where $p(\boldsymbol{\theta})$ is the prior density, and $\mathbb{1}[d(\boldsymbol{S}(\boldsymbol{\theta}^c), \boldsymbol{S}^R) \leq h]$ is an indicator whether the simulation outcome falls within the tolerance radius.

4 Applications

4.1 A price learning mechanism

As a testbed for the application of some of the Bayesian techniques described above, we consider the stock market model proposed in Cliff and Bruten (1997) and used by Grazzini and Richiardi (2015). There is an order book where traders can ask or bid. Traders only observe the book and know nothing about the demand and supply schedule; they cannot lend or borrow. The limit price of agent i in period t is

$$p_{it} = v_i(1 + \mu_{it}) \tag{15}$$

where v_i is the (constant) value of the traded asset for agent i and μ_{it} is a profit margin (positive for sellers and negative for buyers). In each period traders look at the book and define a target price τ_{it}: traders increase their target price if the last trade occurred at a high price, and lower

it otherwise. The actual change in the limit price is:

$$\Delta_{it} = \beta_i(\tau_{it} - p_{it}) \tag{16}$$

where β_i is a behavioural parameter determining how traders react to a difference between the target price and the current price p_{it}. The profit margin is updated as follows:

$$\mu_{i,t+1} = \frac{p_{it} + \Delta_{it}}{v_i} - 1 \tag{17}$$

and determines the updated limit price, $p_{i,t+1}$. For simplicity, it is assumed that $\beta_i = \beta, \forall i = 1, \ldots, N$.

We perform Monte Carlo experiments where the pseudo-true data are generated, as in Grazzini and Richiardi (2015), with $\beta = 0.55$ and $N = 11$ buyers and sellers. We adopt a uniform prior in [0,1]. We first estimate the model by KDE, with 'brute force' sampling of the parameters space. Second, we replace grid exploration with MCMC sampling. We then turn to parametric estimation of the likelihood, by considering augmentation of the model with Gaussian errors. Finally, we experiment with a simple rejection sampling ABC algorithm to embed the MSM strategy of Grazzini and Richiardi (2015) in a Bayesian framework.

4.2 Kernel density estimation

The model is simulated, for every value of the parameter β, for 1,500 periods, once the long-run stationary state has been reached. We employ a Gaussian kernel, with optimal bandwith $h = 1.06\hat{\sigma}S^{-0.2}$, where $\hat{\sigma}$ is the estimated standard deviation in the simulated sample of size S (Silverman, 1986, p. 45). Figure 1 depicts the (normalized) posterior obtained for one series of pseudo-true data, computed on 100 equispaced values of β in the interval (0,1), for different numbers of simulated trading days.

The posterior shrinks as the number of observations increases, though it converges to a value slightly below 0.55. This is the effect of the bias introduced by KDE, and goes in an *a priori* unknown direction.[16] Experiments with different random seeds confirm the presence of a (downward) bias (figure 2).

[16] By converse, the small sample bias associated with MSM, arising from non-linearities of the moment functions, is of predictable direction (see Grazzini et al., 2012). Andrieu and Roberts (2009) propose a 'pseudo-marginal approach' to stochastic simulation in a MCMC Bayesian framework with an unbiased estimator of the likelihood.

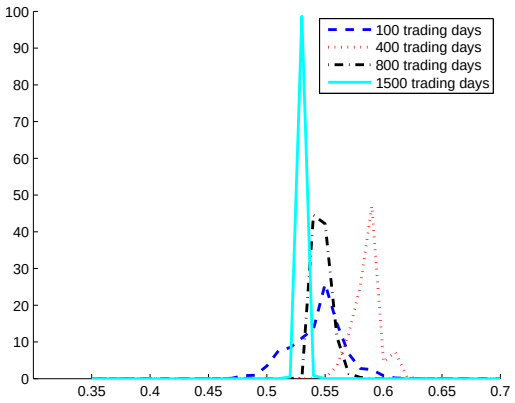


Figure 1: Posterior distribution of the learning parameter β with kernel density estimation and grid exploration of the parameter space (100 equispaced values in [0,100]). Different numbers of trading days are assumed to be observed. The pseudo-true value is $\beta = 0.55$.

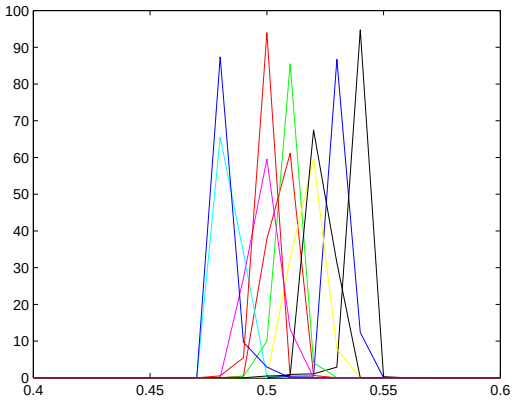


Figure 2: Posterior distribution of the learning parameter β, kernel density estimation, grid exploration with 100 equispaced values in [0,100], 1,500 observed periods, ten different random seeds. True value is $\beta = 0.55$.

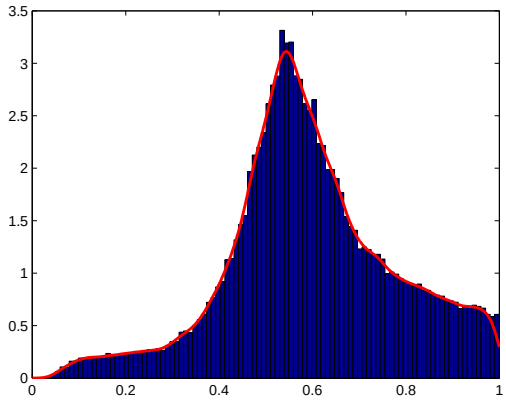


Figure 3: Posterior distribution of the learning parameter β with kernel density estimation of the likelihood and MCMC sampling. 400,000 simulations of 1,500 trading days each are performed. 1,500 pseudo-true trading days are assumed to be observed. The pseudo-true value is $\beta = 0.55$.

As for what concerns performance, each simulation lasting 1,500 periods takes 7.2 secs to run on our computer.[17] For each simulation, KDE takes an additional 6.2 secs. As we have see, grid exploration, for the chosen density of the grid, requires 10,000 simulations (each with its own KDE). Total computing time is therefore $10,000 \times (7.2 + 6.2)$ secs $= 37.2$ hours.

Grid exploration performs very well with a small number of parameters (only one, in our case), as no simulations are 'wasted'. However, as the number of parameters increases, grid exploration becomes infeasible. We therefore experiment with a random walk Metropolis-Hastings MCMC sampling scheme. We perform 400,000 simulations, each lasting 1,500 periods, using 40 parallel processes each lasting 10,000 simulations, discarding the first 1,000 simulations as burn-in. For each simulation, KDE is performed. Figure 3 reports the posterior, using the same random seed as in the 'brute force' grid exploration.

The posterior is much more dispersed than under 'brute force' grid exploration; at the same time, computing time increases by a factor of 40 (from 10,000 to 400,000 simulations), requiring parallelisation. The reason why MCMC sampling is much less efficient when grid exploration is feasible (i.e. with a small number of parameters) is that all simulations are used in grid exploration to produce the density in fig. 1, while most of the simulations end up being discarded. On the other hand, the number of simulations required to achieve a specific accuracy grows much slower with the number of parameters than in grid exploration.

[17] A Dell Precision R7910 with two 2.5 GHz Intel Xeon CPU E5-2680 v3 processors (each with 12 cores and 24 threads) and 128 GB of RAM, available at the Complexity Lab in Economics at the Department of Economics and Finance at the Catholic University of Milan.

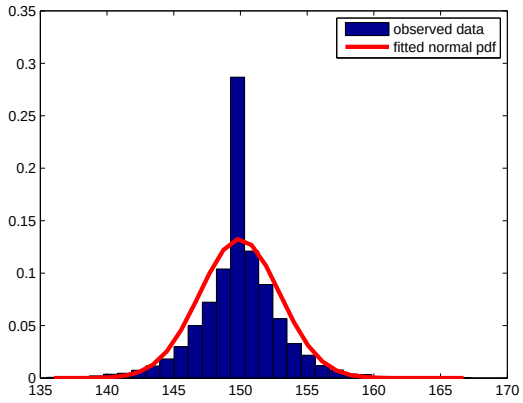


Figure 4: Price distribution in the stationary state, $\beta = 0.55$.

4.3 Augmentation with external errors

As we have seen, computational costs in Bayesian estimation of AB models come from three sources: (i) the time required for running one simulation, (ii) the time required for obtaining an estimate of the likelihood function, given the simulation outcome, and (iii) the number of times that steps (i) and (ii) have to be repeated. We assume that the coding of the model is already efficient, hence simulation time cannot be reduced.[18] MCMC techniques, or other efficient sampling methods, are the only feasible options when the number of parameters is not small, but they perform poorly, with respect to 'brute force' grid exploration, in low-dimensional spaces: hence, gains have to be searched elsewhere.

We then go parametric and assume Gaussian noise, as described in section 3.2. This assumption finds some support in the simulated data, which however are much more concentrated around the theoretical equilibrium value of $p = 150$ (figure 4).

In spite of this discrepancy, computational experience with the new posterior suggests that it performs very well. In figure 5 we present the new posteriors using grid exploration, which quite understandably are smoother than those obtained under KDE, have only a slightly bigger variance, and are more centered on the true value of the parameter (smaller bias).

Finally, figure 6 reports the posterior distribution of the learning parameter β with Gaussian density estimation and MCMC sampling, based on 400,000 simulations of 1,500 periods each, with a burn-in phase of 1,000 simulations, running in parallel on 40 different cores.

As the assumption of normality is not too bad a description of the distribution of the

[18] An interesting approach, which is related to the normality assumption made in this section but is not further investigated here, is using Gaussian process emulators to approximate the outcomes of the simulation model without actually performing all the simulations (O'Hagan, 2006; Bijak et al., 2013).

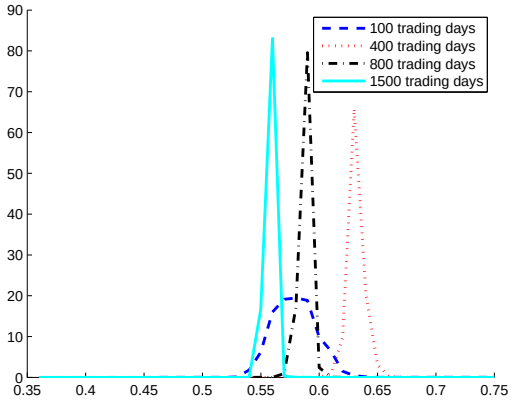


Figure 5: Posterior distribution of the learning parameter β with Gaussian density estimation and grid exploration of the parameter space (100 equispaced values in [0,100]). Different numbers of trading days are assumed to be observed. The pseudo-true value is $\beta = 0.55$.

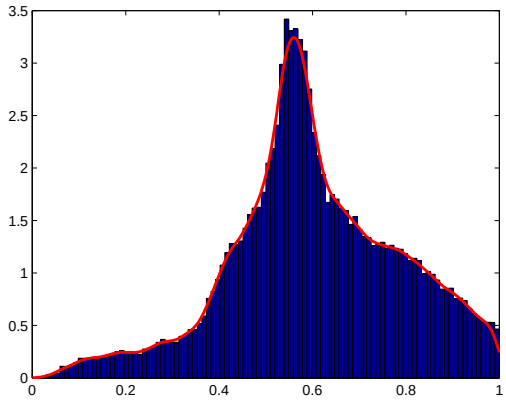


Figure 6: Posterior distribution of the learning parameter β with Gaussian density estimation of the likelihood and MCMC sampling. 400,000 simulations of 1,500 trading days each are performed. 1,500 pseudo-true trading days are assumed to be observed. The pseudo-true value is $\beta = 0.55$.

equilibrium price around its stationary state, the posterior distribution look similar to the one obtained under KDE (figure 3).

The gains in terms of computing time are, on the other hand, impressive: estimation under the normality assumption is practically instantaneous (it only takes 0.014 sec per simulation), compared to 6.2 secs per simunlation for KDE. Given that KDE accounted for almost half of total computing times, both with grid exploration and with MCMC sampling, computing costs are therefore also almost halved.

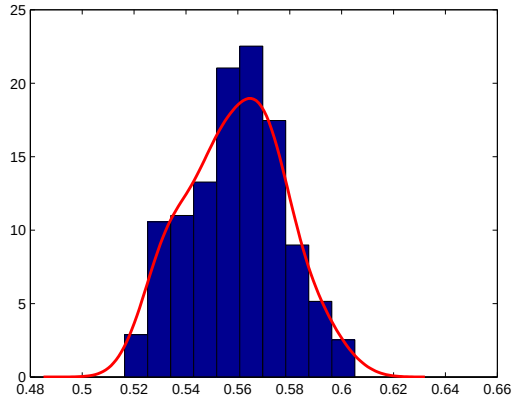
4.4 Approximate Bayesian Computation

Our last exercise considers ABC techniques. Here, we use as our summary statistics the standard deviation of the price, that we know from Grazzini and Richiardi (2015) discriminates well between different values of the parameter. We apply the most basic rejection sampling ABC algorithm. Figure 7 reports the posterior distributions obtained with 400,000 simulations of 1,500 periods each (parallelised on 40 different cores), for different values of the tolerance threshold h.

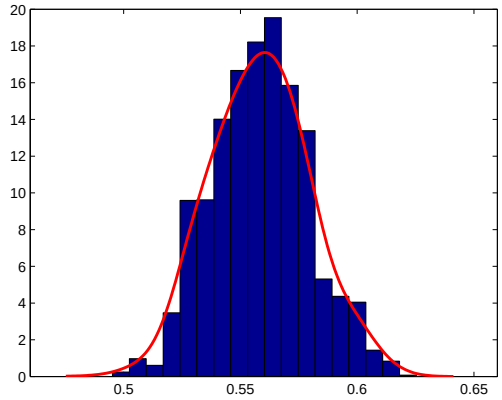
The trade-off between more accuracy but less precision, for small values of h, and less accuracy but more precision, for bigger values of h, is evident, with an intermediate level of $h = 0.1$ achieving a good compromise between the number of accepted parameterisations, and the ability to shape a posterior distribution out of the Gaussian prior. Computing time for the rejection sampling ABC algorithms is also almost nil, so total time is approximately the same as with Gaussian density estimation and MCMC sampling.

5 Conclusions

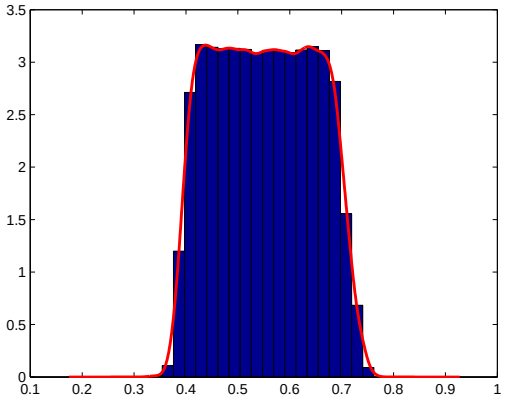
In this paper we have described how simple Bayesian techniques can be applied to the estimation of AB models, as an alternative to the SMD techniques. Lack of analytical relationships between aggregate variables in AB models implies that the likelihood function has to be estimated by simulation, and cannot be functionally approximated around a steady state. Moreover, because individual behaviour in AB models is not expressed in terms of equilibrium (i.e., rational expectations), the steady state can only be found, generally speaking, by running the model. This implies that the computational burden of these techniques is increased vis-á-vis that of analytical models. This problem is exacerbated by the fact that AB models are generally



$h = 0.01$



$h = 0.1$



$h = 1$

Figure 7: Posterior distribution of the learning parameter β with likelihood-free ABC estimation and rejection sampling. 400,000 simulations of 1,500 periods each are performed. 1,500 pseudo-true periods are assumed to be observed. Different values of the tolerance threshold are considered (the standard deviation of price, used as summary statistics, ranges approximately from 0 to 5). The pseudo-true value is $\beta = 0.55$.

complicated models with many equations, which run slow.

In our testbed model we have compared three approaches to Bayesian estimation, namely non-parametric estimation of the likelihood, parametric (Gaussian) estimation of the likelihood, and likelihood-free Approximate Bayesian Computation. We also experimented with different sampling schemes, from simple 'brute force' grid exploration to more sophisticated Markov chain Monte Carlo and rejection sampling algorithms.

Our findings can be summarised as follows. First, simulation time can be a major obstacle to estimating large-scale AB models. Even in our extremely simple model, with one parameter only, simulation time accounts for more than 50% of all estimation time. Second, in simple models grid exploration is by far more efficient than alternative 'efficient sampling' schemes that filters proposed values for the parameters to arrive at a sample of values drawn from the posterior distribution, the reason being that grid exploration of the parameters' space does not 'throw away' anything. Third, both parametric estimation of the likelihood and likelihood-free ABC methods allow for a significant reduction in computing times. However, when ABC methods rely on 'good' summary statistics, they perform better.

Table 1 summarises the performance of the different techniques, in our sample model.

Inferential procedure	Sampling	Simulations	Total time	Qualitative assessment
Non-parametric KDE	Grid exploration	10,000	37 h	Very good precision, small bias
Parametric Gaussian	grid exploration	10,000	20 h	Very good precision
Non-parametric KDE	MCMC	400,000	1,488 h	Poor precision
Parametric Gaussian	MCMC	400,000	800 h	Poor precision
ABC	Rejection sampling	400,000	800 h	Good precision

Table 1: Performance of different Bayesian techniques. Running one simulation of 1,500 periods (trading day) requires 7.2 secs on our reference machine. Performing KDE requires 6.2 secs per simulation. Gaussian density estimation and ABC require practically no additional costs.

These results can provide insights on common issues in AB modelling, but are specific to the model being tested. In particular, it would be interesting to analyse the performance of the different techniques in a large-scale AB model, something we leave for future research.

Our final remark concerns ABC. A feature of Bayesian estimation is not requiring the choice of summary statistics. Recourse to ABC obviously undermines this, and is subject to the same pros and cons of SMD techniques. Moments selection is generally left to the intuition of the researcher, and then validated by sensitivity analysis on the model behaviour; however, in complicated models finding the right moments could be a serious challenge. On the other hand,

the need to analyse the behaviour of the summary statistics prior to estimation prompts a better understanding of the model behaviour, and can lead to more robust estimation.[19]

Acknowledgements. We thank Christian Robert for introducing us to ABC. We are indebted to seminar participants at the University of Lancaster and to the participants to the workshop "Agent-Based and DSGE Macroeconomic Modelling: Bridging the Gap" at the University of Surrey –to Alessandro Gobbi in particular– for their comments. Jakob Grazzini gratefully acknowledges the support by the European Union, Seventh Framework Programme FP7/2007-2013 Socio-economic Sciences and Humanities under Grant Agreement No. 612796 MACFINROBODS. Matteo Richiardi gratefully acknowledges support by a Marie Curie Intra European Fellowship within the 7th European Community Framework Programme.

[19]This has also been noted in DSGE modelling. Ruge-Murcia (2007) compares SMD to Bayesian techniques and concludes that moment-based estimation methods compare very favourably to the more widely used likelihood-based methods, being more robust to misspecification and less affected by stochastic singularity.

References

Andrieu, C., de Freitas, N., Doucet, A., and Jordan, M. (2003). An introduction to mcmc for machine learning. *Machine Learning*, 50(1):5–43.

Andrieu, C. and Roberts, G. O. (2009). The pseudo-marginal approach for efficient monte carlo computations. *Annals of Statistics*, 37(2):697–725.

Arulampalam, M., Maskell, S., Gordon, N., and Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50:174–188.

Banish, S., Lima, R., and Araújo, T. (2012). Aggregation and emergence in agent based models: A markov chain approach. Working Paper WP 25/2012/DE/UECE, School of Economics and Management, Technical University of Lisbon.

Beaumont, M. A., Zhang, W., and Balding, D. J. (2002). Approximate bayesian computation in population genetics. *Genetics*, 162(4):2025–2035.

Bernardo, J. (1997). Noninformative priors do not exist: A discussion. *Journal of Statistical Planning and Inference*, 65(159-189).

Bijak, J., Hilton, J., Silverman, E., and Cao, V. D. (2013). Reforging the wedding ring: exploring a semi-artificial model of population for the united kingdom with gaussian process emulators. *Demographic Research*, 29(27):729–766.

Cliff, D. and Bruten, J. (1997). Minimal-intelligence agents for bargaining behaviors in market based environments. *HP Laboratories Bristol*, (HPL-97-91).

Fearnhead, P. and Prangle, D. (2012). Constructing summary statistics for approximate bayesian computation: semi-automatic approximate bayesian computation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(3):419–474.

Fisher, R. (1922). On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society A*, 222:309–368.

Fu, Y. and Li, W. (1997). Estimating the age of the common ancestor of a sample of dna sequences. *Molecular biology and evolution*, 14(2):195–199.

Grazzini, J. and Richiardi, M. (2015). Consistent estimation of agent-based models by simulated minimum distance. *Journal of Economic Dynamics and Control*, 51:148–165.

Grazzini, J., Richiardi, M., and Sella, L. (2012). Small sample bias in msm estimation of agent-based models. In Andrea Teglio, Simone Alfarano, E. C.-C. M. G.-V., editor, *Managing Market Complexity. The Approach of Artificial Economics.*, Lecture Notes in Economics and Mathematical Systems. Springer.

Hartig, F., Calabrese, J. M., Reineking, B., Wiegand, T., and Huth, A. (2011). Statistical inference for stochastic simulation models theory and application. *Ecology Letters*, 14:816–827.

Holmes, W. R. (2015). A practical guide to the probability density approximation (pda) with improved implementation and error characterization. *Journal of Mathematical Psychology (forthcoming)*.

Kemeny, J. and Snell, J. (1976). *Finite Markov Chains*. Springer, New York.

Marin, J.-M., Pudlo, P., Robert, C. P., and Ryder, R. J. (2011). Approximate bayesian computational methods. *Statistics and Computing*, 22(6):1167–1180.

O'Hagan, A. (2006). Bayesian analysis of computer code outputs: A tutorial. *Reliability Engineering & System Safety*, 91(1290-1300).

Prangle, D. (2014). Lazy abc. *Statistics and Computing*, pages 1–15.

Richiardi, M. (2012). Agent-based computational economics. a short introduction. *The Knowledge Engineering Review*, 27(2):137–149.

Ruge-Murcia, F. (2007). Methods to estimate dynamic stochastic general equilibrium models. *Journal of Economic Dynamics and Control*, 31(2599-2636).

Silverman, B. (1986). *Density Estimation for Statistics and Data Analysis*. Chapman and Hall.

Sisson, S. A., Fan, Y., and Beaumont, M., editors (2016). *Handbook of Approximate Bayesian Computation*. Taylor & Francis.

Sun, D. and Berger, J. (2006). Objective bayesian analysis for the multivariate normal model. In *Proc. Valencia / ISBA 8th World Meeting on Bayesian Statistics, Benidorm (Alicante, Spain), June 1st-6th.*

Tavaré, S., Balding, D. J., Griffiths, R. C., and Donnelly, P. (1997). Inferring coalescent times from dna sequence data. *Genetics*, 145:505–518.

Turner, B. M. and Sederberg, P. B. (2013). A generalized, likelihood-free method for posterior estimation. *Psychonomic Bullettin & Review*, 20(5).

Turner, B. M. and Zandt, T. V. (2012). A tutorial on approximate bayesian computation. *Journal of Mathematical Psychology*, 56:69–85.

Wood, S. N. (2010). Statistical inference for noisy nonlinear ecological dynamic systems. *Nature*, 466:1102–1107.

A Gaussian density estimation

In this Appendix we derive the expression for the likelihood and the posterior density functions under the assumption of Gaussian errors, either for $\boldsymbol{\epsilon}$ (eq. 10) of for $\mathbf{u}$ (eq. 11). Under the Gaussian assumption, disturbances are distributed as $\mathscr{N}_K(\mathbf{0}, \boldsymbol{\Sigma})$, where K is the dimensionality of $\mathbf{y}$, the number of aggregate observables that the model is able to reproduce. We can either estimate the free elements of matrix $\boldsymbol{\Sigma}$ or assume that it is diagonal with diagonal elements $\sigma_{kk},\ k = 1, ..., K$ which can be thought of as fixed (to some small value) or unknown. In the stationary state, we then have that

$$\mathbf{y}_{t+1} \sim \mathscr{N}_K\left(\mathbf{g}^*(\boldsymbol{\theta}), \boldsymbol{\Sigma}(\boldsymbol{\theta})\right). \tag{18}$$

or, in the case of disturbances added to the data,

$$\mathbf{y}^R_{t+1} \sim \mathscr{N}_K\left(\mathbf{g}^*(\boldsymbol{\theta}^R), \boldsymbol{\Sigma}(\boldsymbol{\theta})\right). \tag{18'}$$

The joint distribution of the observables at time t, in turn, is given directly as follows:

$$p\left(\mathbf{y}^R_t | \boldsymbol{\Xi}_t, \boldsymbol{\theta}, \boldsymbol{\Sigma}(\boldsymbol{\theta})\right) \propto |\boldsymbol{\Sigma}(\boldsymbol{\theta})|^{-1/2} \exp\left\{-\tfrac{1}{2}\left[\boldsymbol{y}^R_t - \mathbf{g}^*(\boldsymbol{\theta})\right]' \boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\left[\boldsymbol{y}^R_t - \mathbf{g}^*(\boldsymbol{\theta})\right]\right\}. \tag{19}$$

from which a likelihood function can be easily derived:[20]

$$\mathscr{L}\left(\boldsymbol{\theta}, \boldsymbol{\Sigma}; \mathbf{Y}^R\right) \propto |\boldsymbol{\Sigma}|^{-T/2} \exp\left\{-\tfrac{1}{2}\mathrm{tr}\left[\mathbf{A}(\boldsymbol{\theta})\boldsymbol{\Sigma}^{-1}\right]\right\} \tag{20}$$

where $\mathbf{A}(\boldsymbol{\theta}) = \sum_{t=1}^{T}\left[\boldsymbol{y}^R_t - \mathbf{g}^*(\boldsymbol{\theta})\right] \cdot \left[\boldsymbol{y}^R_t - \mathbf{g}^*(\boldsymbol{\theta})\right]'$.

The likelihood is formed under the assumption that the disturbances are an iid process so that there is, for example, no autocorrelation (the assumption is easy to remove).

Further simplification can be obtained by assuming that the elements of $\boldsymbol{\Sigma}$ are independent from those of $\boldsymbol{\theta}$:

$$p(\boldsymbol{\theta}, \boldsymbol{\Sigma}) = p(\boldsymbol{\theta})p(\boldsymbol{\Sigma}). \tag{21}$$

[20] We use the 'trace trick' for Gaussian likelihoods: $\mathbf{X}'\boldsymbol{\Sigma}^{-1}\mathbf{X} = \mathrm{tr}\left(\boldsymbol{\Sigma}^{-1}\mathbf{X}\mathbf{X}'\right)$. This comes from two properties of the trace: (i) $\mathrm{tr}(AB) = \mathrm{tr}(BA)$ (if all dimensions work out, ie. if AB is a square matrix), (ii) $\mathrm{tr}(c) = c$ if c is a constant. Because $\mathbf{X}'\boldsymbol{\Sigma}^{-1}\mathbf{X}$ is a scalar, we can then write $\mathbf{X}'\boldsymbol{\Sigma}^{-1}\mathbf{X} = \mathrm{tr}\left(\mathbf{X}'\boldsymbol{\Sigma}^{-1}\mathbf{X}\right) = \mathrm{tr}\left(\mathbf{X}\mathbf{X}'\boldsymbol{\Sigma}^{-1}\right)$.

We then assume diffuse priors:

$$p(\boldsymbol{\theta}) = \text{const.} \tag{22}$$

$$p(\boldsymbol{\Sigma}) \propto |\boldsymbol{\Sigma}|^{-(K+1)/2} \tag{23}$$

where $p(\boldsymbol{\Sigma})$ is the commonly used independence-Jeffreys prior (Sun and Berger, 2006), that is, invariant to re-parametrisation of $\boldsymbol{\Sigma}$.

This leads to the following posterior:

$$p\left(\boldsymbol{\theta}, \boldsymbol{\Sigma} | \mathbf{Y}^{\mathbf{R}}\right) \propto |\boldsymbol{\Sigma}|^{-\frac{T+K+1}{2}} \exp\left\{-\tfrac{1}{2}\text{tr}\left[\mathbf{A}\left(\boldsymbol{\theta}\right)\boldsymbol{\Sigma}^{-1}\right]\right\} \tag{24}$$

Given that $\boldsymbol{\Sigma}$ is generally not an object of interest (though it could be used as measure of how well the model fits the data), we can integrate it out analytically to obtain:

$$p\left(\boldsymbol{\theta} | \mathbf{Y}^{R}\right) \propto |\mathbf{A}\left(\boldsymbol{\theta}\right)|^{-T/2} \tag{25}$$

In this way, kernel-based estimation is avoided altogether, saving computational time.

Notice that the new posterior does not depend on the different elements of $\boldsymbol{\Sigma}$. The dimensionality of this matrix is the same as the dimensionality of $\mathbf{Y}^{R}$ which is, in our case, quite low. Generally speaking, it is required that if $\boldsymbol{\Sigma}$ is $K \times K$ we should have $\frac{K(K+1)}{2} << T$.