CHARLES UNIVERSITY
FACULTY OF SOCIAL SCIENCES
Institute of Economic Studies

Simulation-based estimation methods in financial econometrics: Analysis of performance and comparison

Master's thesis

Author: Bc. Ondřej Karlíček
Study program: Economics and Finance
Supervisor: PhDr. Jiří Kukačka, Ph.D.
Year of defense: 2024

Declaration of Authorship

The author hereby declares that he or she compiled this thesis independently, using only the listed resources and literature, and the thesis has not been used to obtain any other academic title.

The author grants to Charles University permission to reproduce and to distribute copies of this thesis in whole or in part and agrees with the thesis being used for study and scientific purposes.

During the preparation of this thesis, the author used `ChatGPT` in order to revise code and check grammar. After using this tool/service, the author reviewed and edited the content as necessary and takes full responsibility for the content of the publication.

Prague, July 30, 2024

Ondrej Karlicek

Abstract

This thesis investigates the performance of simulation-based estimation methods in financial econometrics, specifically focusing on their application to agent-based models. Traditional estimation techniques often fail due to the intractability of analytical solutions in agent-based models, necessitating the use of innovative simulation-based approaches. The study compares two frequentist methods, Simulated Method of Moments (SMM) and Non-parametric Simulated Maximum Likelihood (NPSML), with their Bayesian counterparts, Approximate Bayesian Computation (ABC) and Bayesian Estimation (BE), respectively. On simple benchmark models, such as the AR(2) model and the ARMA(1,1)-GARCH(1,1) model, the simulation-based methods match the performance of traditional techniques. The well-known agent-based model from Franke and Westerhoff (2012) is the main model of interest. The results do not indicate a clear overall winner, as the performance varies parameter by parameter. However, Bayesian methods generally outperform their frequentist counterparts. ABC and SMM provide less biased estimates than the likelihood methods, NPSML and BE. On the other hand, the estimates from NPSML and BE are more stable across different simulation runs. Additionally, this study contributes to the understanding of the behavior of an extended NPSML approach designed to handle latent variables.

JEL Classification	C13, C15, C52, C58, G40
Keywords	Agent-based modelling, Behavioural finance, Simulated method of moments, Simulated maximum likelihood, Bayesian estimation
Title	Simulation-based estimation methods in financial econometrics: Analysis of performance and comparison

Abstrakt

Tato diplomová práce zkoumá výkonnost metod simulovaného odhadu ve finanční ekonometrii, se specifickým zaměřením na jejich aplikaci na multiagentních modelech. Tradiční odhadové techniky často selhávají kvůli neřešitelnosti analytického řešení v multiagentních modelech, což vyžaduje použití inovativních metod založených na simulacích. Studie porovnává dvě frekventistické metody, Simulovanou metodu momentů (SMM) a Neparametrickou simulovanou maximální věrohodnost (NPSML), s jejich bayesovskými protějšky, Přibližný bayesovský výpočet (ABC) a Bayesovský odhad (BE). Na jednoduchých referenčních modelech, jako je AR(2) model a ARMA(1,1)-GARCH(1,1) model, metody založené na simulaci dosahují srovnatelné výkonnosti s tradičními technikami. Hlavním modelem zájmu je známý multiagentní model od Franke a Westerhoff (2012). Výsledky neukazují jasného celkového vítěze, protože výkonnost se liší parametr od parametru. Nicméně, bayesovské metody obecně překonávají své frekventistické protějšky. ABC a SMM poskytují méně zkreslené odhady než metody založené na věrohodnosti, NPSML a BE. Na druhou stranu, odhady z NPSML a BE jsou stabilnější napříč různými simulacemi. Tato studie také přispívá k pochopení chování rozšířeného NPSML, který je navržen pro práci s latentními proměnnými.

Klasifikace JEL	C13, C15, C52, C58, G40
Klíčová slova	Multiagentní model, Behaviorální finance, Simulovaná metoda momentů, Simulovaná metoda maximální věrohodnosti, Bayesovské odhady
Název práce	Simulační metody odhadu ve finanční ekonometrii: Analýza výkonnosti a srovnání

Acknowledgments

I would like to express my deepest gratitude to my supervisor, PhDr. Jiří Kukačka, Ph.D., for his invaluable guidance, support, and encouragement throughout the course of my research. His expertise and insights were crucial to the completion of this thesis. I also want to extend my heartfelt thanks to my family for their unwavering patience and support. Additionally, I am grateful to MetaCentrum for providing the computational resources necessary for this research. Computational resources were provided by the e-INFRA CZ project (ID:90254), supported by the Ministry of Education, Youth and Sports of the Czech Republic.

Typeset in LaTeXusing the IES Thesis Template.

Bibliographic Record
Karlicek, Ondrej: *Simulation-based estimation methods in financial econometrics: Analysis of performance and comparison.* Master's thesis. Charles University, Faculty of Social Sciences, Institute of Economic Studies, Prague. 2024, pages 83. Advisor: PhDr. Jiří Kukačka, Ph.D.

Contents

List of Tables

List of Figures

Acronyms

ABC	Approximate Bayesian Computation
ABM	Agent-Based Model
AR	Autoregressive
ARMA	Autoregressive Moving Average
BE	Bayesian Estimation
DCA	Discrete Choice Approach
GARCH	Generalised Autoregressive Conditional Heteroskedasticity
GMM	Generalized Method of Moments
KDE	Kernel Density Estimation
LIL	Limited Information Likelihood
LSE	LogSumExp
MA	Moving Average
MCMC	Markov Chain Monte Carlo
MLE	Maximum Likelihood Estimator
NPSML	Non-parametric Simulated Maximum Likelihood
OLS	Ordinary Least Squares
SMC	Sequential Monte Carlo
SML	Simulated Maximum Likelihood
SMM	Simulated Method of Moments

Chapter 1

Introduction

Expectations are an integral part of everyday life, as opinions about the future shape our decisions. They are also a cornerstone of economic theory, with most models built on certain assumptions about expectations. These assumptions have evolved throughout history to align with the current state of knowledge and the needs of economic theory.

In the early days of economic research, the dominant assumptions were static or naive expectations (Evans & Honkapohja 2001). Under these assumptions, expectations were either the same as in the past period or extrapolated from past values. However, to improve forecasting accuracy, these expectations were largely replaced by the concept of rational expectations, introduced by Muth (1961). Under rational expectations, individuals form their decisions based on all available information and their understanding of the economic model. This shift to rational expectations has provided an important grounding for many economic models and has significantly influenced economic theory and policy.

In empirical financial data, it is typical to find occurrences of extreme returns (fat tails), periods of clustered volatility, speculative bubbles, and the absence of autocorrelation. These are known as stylized facts (Cont 2001). Traditional techniques based on rational expectations often fail to replicate these stylized facts (Hong & Stein 1999). This has led researchers to focus on heterogeneous expectations. A prominent approach in this shift is the use of financial Agent-Based Models (ABMs). ABMs simulate the interactions between boundedly rational agents, creating unique dynamics that can replicate and explain stylized facts (Cont 2007). Allen & Taylor (1990) and Frankel & Froot (1990) demonstrate that financial market behavior is mainly driven by two types of agents: fundamentalists, who expect mispricing to correct to fundamental values, and chartists, who trade based on recent trends. These two types of agents are the cornerstone of most financial ABMs. The dynamics of the model are driven by the interactions and strategy-switching between these agents (Brock & Hommes 1997; Alfarano *et al.* 2008; Franke & Westerhoff 2012).

The correct estimation of financial ABMs is a major challenge they face in order to be widely adopted in practical applications (Lux & Zwinkels 2018). The analytical solutions of ABMs are often intractable, preventing the use of traditional estimation techniques. Therefore, researchers must apply novel methods to estimate the parameters, with most attention given to simulation-based methods. Recent literature has focused more on introducing and developing new approaches, creating a gap in the literature devoted to the comparison and benchmarking of current methods, as their theoretical properties are still not well understood (Grazzini *et al.* 2017; Platt 2020). Further comparison can provide researchers with more insight into the current methods, allowing them to make better decisions about when to use each method.

This thesis builds on the comparison work of Platt (2020) by incorporating new methods and different types of benchmark models. Following Platt (2020), we utilize the most popular method among financial ABMs, the Simulated Method of Moments (SMM), but with new insights from Zila & Kukacka (2023). Additionally, we introduce a recently promising method, the Non-parametric Simulated Maximum Likelihood (NPSML), which has shown encouraging results in Kukacka & Barunik (2017). The emergence of Bayesian techniques in the literature motivates us to include Bayesian counterparts to these two methods. Specifically, we incorporate the likelihood-based Bayesian Estimation (BE) from Zhang *et al.* (2023), which approximates the posterior distribution using likelihoods obtained via NPSML. A variant of this approach is the winning method in Platt (2020). Finally, we include the likelihood-free method Approximate Bayesian Computation (ABC) inspired by Lux (2023a), which employs moment matching similar to the SMM and is popular among biological ABMs. Thus, we utilize simulated versions of the popular frequentist methods of moments and maximum likelihood, along with their Bayesian counterparts. We extend Platt (2020) by incorporating two new methods, while slightly modifying the remaining two based on recent findings.

For the benchmark models, our primary focus is on models with latent variables, extending the work of Platt (2020), where the presence of latent variables is debatable. We begin with a simple Autoregressive (AR) model, which provides a baseline every method should meet. The second model is an Autoregressive Moving Average (ARMA)-Generalised Autoregressive Conditional Heteroskedasticity (GARCH) model. This traditional model, with its latent variable, offers much more complex dynamics. Both models are typically estimated using traditional econometric tools, allowing us to compare the performance of our methodologies with these conventional approaches. The last model is our main interest: the Franke & Westerhoff (2012) model, which is not feasible for traditional methods. This model, popular among financial ABMs, examines the interaction between fundamentalists and chartists in the financial market, with a switching mechanism driven by the dynamics of a latent variable. Models with latent variables provide an interesting framework to test our methods. To our knowledge, the NPSML method has been utilized for financial ABMs with latent variables only as part of Bayesian estimation by Platt (2022). Therefore, we aim to shed more light on its performance in this context, including its application in a frequentist framework.

The thesis is structured as follows: Chapter 2 presents the historical development inside the financial ABM framework and the recent literature. The presented methods' theoretical properties and implementation are in Chapter 3. Chapter 4 describe the benchmark models in more detail and state their parametrization. The settings of the overall comparison study, the individual methods, and the optimization constraints are described in Chapter 5. Chapter 6 provides the results and assesses the methods' performance. Chapter 7 continues with the sensitivity analysis and further possible improvements. Finally, the conclusion of the thesis is in Chapter 8.

Chapter 2

Literature Review

The chapter presents the literature to delve more into the historical evolution of the financial ABMs and the current state of the research. Firstly, Section 2.1 surveys the development of the ABMs in the financial environment. Subsequently, Section 2.2 lists the estimation techniques used within the financial ABMs as the correct estimation of the parameters is the main difficulty for the empirical use of ABMs.

2.1 Financial Agent-Based Models

The central assumption of economic research is a fully rational agent (Hirshleifer 1985). However, the rationality assumption within the models limits the resulting time series. This assumption does not explain the stylized facts of the financial markets (Cont 2001). For the empirical financial data, it is typical to find occurrences of volatility clustering or fat-tailed returns (Ding *et al.* 1993; Fama 1965).

The alternative can be financial ABMs, which explain the financial markets by the interaction of the individual agents, such as traders and market makers. The agents do not act under the rationality assumption. Instead, they are characterized by bounded rationality. Zeeman (1974) conducts one of the pioneering studies in the financial ABMs using the catastrophe theory to the unstable asset price returns. The agents in this model are divided into groups deciding based on the fundamental value or the recent trend in price changes. This template can be seen even in recent models. Bornholdt (2001) reproduces clustered volatility and fat-tails with the Ising model, where buyers and sellers are put on the grid, and the agents switch based on probability conditional on their neighbors.

Two types of expectations can control the dynamics of financial markets (Allen & Taylor 1990; Frankel & Froot 1990). Fundamentalists base their expectations on the belief that any mispricing will eventually be corrected to align with the fundamental value. In contrast, chartists justify their decisions by predicting that the current trend will persist in the short term. These two strategies allow for constructing a 2-type model or N-type model with one fundamentalist strategy and many chartist strategies.

Further important developments are conducted by Kirman (1993) and Brock & Hommes (1997). Kirman (1993) utilizes an ant type of system; the agents switch between the strategies based on the behavior of other agents. An important feature is a herding mechanism; the probability of switching is influenced by the proportion of agents in each state (strategy). Alfarano *et al.* (2005; 2008) generalizes the approach by introducing a different probability of switching to each group. The ant type of system models can replicate the stylized fact of financial time series.

Brock & Hommes (1997) take another approach, guiding the transition through the adaptive belief system. The agents select one of the available strategies using a metric that evaluates each based on past performance, prediction errors, or profits. The intensity of the choice parameter influences the overall willingness to switch. Gaunersdorfer & Hommes (2007) extend the former model by incorporating nonlinear feedback mechanisms in the agents' strategy-switching process.

A popular model established by Franke & Westerhoff (2012) is inspired by both philosophies of switching systems. However, it is much closer to the adaptive belief system. The paper implements two switching regimes. The transition probability approach incorporates individual probabilities of switching to each strategy group. The discrete choice approach is not different from the one in an adaptive belief system. Franke & Westerhoff (2012) implement four factors influencing the switching regimes in both approaches: herding, wealth, predisposition, and misalignment. After calibration and comparing the empirical time series, the winning model is constructed using a discrete choice approach with herding, predisposition, and misalignment.

2.2 Estimation Techniques

The analytical solution of the ABMs is usually not feasible, forbidding traditional estimation techniques. However, in a few examples, it is possible to utilize the classical methods, such as least squares, method of moments, or Maximum Likelihood Estimator (MLE). The catastrophe model by Zeeman (1974) is solvable and obtains the closed-form solution using the method of moments (Cobb 1981). Boswijk *et al.* (2007) successfully employ the nonlinear least squares on asset pricing model inspired by Brock & Hommes (1997); another example is ABM for the British pound during the European monetary crisis (De Jong *et al.* 2009). The Ising model by Bornholdt (2001) and the ant-type system model (Alfarano *et al.* 2005; 2008) are analytically solved utilizing the MLE.

More complex ABMs do not have a closed-form solution. Therefore, the researchers need to apply novel methods. The most used is the SMM invented by the McFadden (1989) and Pakes & Pollard (1989). They also derive the asymptotic properties of this method. One of the initial applications in the ABMs field is the simplified version of the Kirman (1993) model (Gilli & Winker 2003). The series of papers by Franke & Westerhoff (2011; 2012; 2016), which focus on developing an already stated model and studying its statistical properties, also employ the SMM with the development of the weighting matrix to better capture the covariances between given moments. Chen & Lux (2018) study the convergence of the SMM to the Generalized Method of Moments (GMM) on the Alfarano *et al.* (2008) model and conduct that due to the limited number of sensible moments for a univariate asset-pricing model, the convergence requires a large sample size. The SMM is sensitive to the moment set selection; thus, Zila & Kukacka (2023) implement moment selection extension to the SMM based on machine learning techniques with the success of achieving a higher performing moment set. Their contribution is also an extension to the weighting matrix from Franke & Westerhoff (2012) with the addition of the overlapping blocks to the bootstrap, which allows the construction of a higher number of the bootstrapping blocks.

The recent rivaling method is NPSML, where the likelihood function is approximated using the simulations from the model, and Kernel Density Estimation (KDE) methods. Kristensen & Shin (2012) develop the NPSML with its theoretical properties, and demonstrate the method on the dynamic models without an analytical solution. The NPSML was firstly utilized in ABMs by Kukacka & Barunik (2017) on the Brock & Hommes (1997) with extensive study of its properties and the sensitivity to the different specifications of the noise term, their empirical application then delve insight into the slight dominance of the fundamentalists in the world's markets. Kukacka & Sacht (2023) further generalize the method for the multivariate macroeconomics model with adjustments in the KDE. The NPSML has an advantage over the SMM in that it does not rely on the preselected set of moments by the research. However, it depends on the assumption that the distribution of the noise term is known beforehand.

The extensive part of the literature is dedicated to the Bayesian techniques with the benefit that the estimation yields the posterior distribution instead of the point estimate. Grazzini *et al.* (2017) pioneer usage of the Bayesian techniques, with the posterior being derived using grid search or the Markov Chain Monte Carlo (MCMC). Both variants are fed with a parametric or non-parametric approximation of the likelihood. For simple single-parameter estimation, the parametric approach with grid search performs better; however, the grid search became overwhelming with more parameters. Therefore, Grazzini *et al.* (2017) recommend for the empirical usage the MCMC algorithm with the non-parametric likelihoods as the parametric approach depends on the specification given by the researcher. Gatti & Grazzini (2020) extend the method by adding information about the time structure of the time series. Another example of the MCMC is in the study from Bertschinger & Mozzhorin (2021) and Lux (2022).

However, the MCMC tends to get stuck in local mode (Zhang *et al.* 2023). Lux (2018) explores the use of the Sequential Monte Carlo (SMC) in the financial ABMs as the more robust algorithm to local modes. SMC is employed to approximate the conditional densities entering the likelihood calculation. Lux (2023b) follows up on his work by implementing two versions of SMC, one to capture the hidden state system to approximate the likelihood and the second as the posterior sampler. The SMC proves to be computationally less demanding than the MCMC within the ABMs. Zhang *et al.* (2023) combine the SMC as the posterior sampler with the non-parametric likelihood approximation approach from Kristensen & Shin (2012) on the macroeconomical ABM, their simulation study conducts that the SMC can deal with multimodal likelihood.

Recently, the usage of the Bayesian likelihood-free methods is being studied in the financial ABMs framework; this approach is popular in the biological ABMs. Lux (2023a) compares variations of the ABC on the Alfarano *et al.* (2008)

model. The ABC is implemented with a rejection sampler and three different versions of the SMC. ABC instead of the likelihood relies on the approach similar in SMM. Based on the numeral simulation exercises, Lux (2023a) concludes that ABC seems more efficient with the information obtained from moments than SMM, and it can be used for sample sizes, which are difficult for numerical likelihood methods. ABC is also briefly studied in Grazzini *et al.* (2017), with the rejection sampler achieving comparable results with the likelihood approaches.

Emerging machine-learning techniques find their place in the study from Platt (2022). The Bayesian estimation is extended with likelihood approximation using neural networks and is able to outperform the approach given by Grazzini *et al.* (2017) on the Brock & Hommes (1997) model and the Franke & Westerhoff (2012) model. Surrogate modeling is another promising machine learning approach. Lamperti *et al.* (2018) utilize the XGBoost algorithm as the meta-model, which learns the relationship between the model's parameters and its output to be then used on the model estimation; the performance is studied on the Brock & Hommes (1997) model.

Despite the rapid emergence of new techniques, the accurate estimation of the financial ABMs remains the main challenge in the field. Recent literature mainly focuses on introducing novel techniques, with a noticeable gap in studies that compare current methods to explore their advantages and disadvantages in depth (Grazzini *et al.* 2017; Platt 2020). An exception to this trend is the study by Platt (2020), which benchmarks the SMM with two similar methods, where approaches from the information theory are utilized instead of the difference between moments. The SMM and their variants are being employed with two different optimization algorithms, particle swarm optimization and a surrogate model method. Additionally, Platt (2020) implements the Bayesian estimation with non-parametric likelihood with the MCMC from Grazzini *et al.* (2017). The performance of these methods is compared using three traditional simple time-series models and the Brock & Hommes (1997) model, with the Grazzini *et al.* (2017) approach demonstrating the best performance.

Chapter 3

Methodology

This chapter presents the estimation techniques employed in this study and their implementation. We begin with two frequentist methods: SMM in Section 3.1 and NPSML in Section 3.2. Subsequently, we transition to Bayesian techniques, Section 3.3 starts with likelihood-free method ABC, which shares similarities with SMM. Finally, Section 3.4 concludes this chapter with BE, which relies on the likelihood approximation.

3.1 Simulated Method of Moments

SMM, also known as the Method of Simulated Moments in the literature, is a historically popular choice for estimating economic ABMs. SMM extends GMM to be applicable in situations lacking a closed-form solution. In cases where there is no analytical or numerical solution to the moments, GMM is inapplicable. While SMM applies to a broad family of models, it is important to note that incorporating simulation introduces a high computational cost, making it suitable only for scenarios where traditional techniques cannot be employed.

SMM beginnings can be found in McFadden (1989) and Pakes & Pollard (1989), who develop the method and demonstrate its consistency and asymptotic normality under the law of large numbers, provided that the simulation errors are independent across observations. Subsequent modifications by Lee & Ingram (1991) and Duffie & Singleton (1990) adapt the method for time series and panel data models, respectively, by outlining its asymptotic properties under additional constraints on error terms specific to these model types.

The core concept involves substituting the analytical or numerical solution of moments by simulation. In statistics, a moment is a quantitative measure that provides information about the shape of a probability distribution. Moments can describe various aspects of the distribution, such as the mean (the first moment) and the variance (the second moment). Additionally, moments can also reflect distributional dependencies over time, such as autocorrelation. In order to achieve an approximation of the moments, it is necessary to conduct multiple runs from the model for given parameter set. The objective is to match the empirical and simulated moments as accurately as possible by running the simulation for a range of parameter sets and minimizing the weighted distance between the two sets of moments to obtain the point estimate. Confidence intervals for the estimates are typically obtained through Monte Carlo simulations, which generate numerous point estimates to calculate standard errors.

3.1.1 Formal Definition

In the formal definition, we follow the approach outlined by Zila & Kukacka (2023). Let us define an empirical time series:

$$\{y_t\}, t = 1, \dots, T_{emp},$$

which can be organized in a column vector $\boldsymbol{y}$.

It is crucial to choose the appropriate moments for a given model and problem. Let us suppose we have selected D moments. We can then approximate a given moment using the function $m_d(\boldsymbol{y})$, where $d = 1, \dots, D$, which computes the sample counterparts. For instance, the second-order auto-correlation moment can be expressed as:

$$m_d(\boldsymbol{y}) = \frac{\sum_{t=2}^{T}(y_t - \bar{y})(y_{t-1} - \bar{y})}{\sum_{t=1}^{T}(y_t - \bar{y})^2}.$$

Here, $\bar{y}$ denotes the sample mean.

Now, we can define an empirical moment vector:

$$\boldsymbol{m}^{emp} = \boldsymbol{m}(\boldsymbol{y}) = [m_1(\boldsymbol{y}), \dots, m_D(\boldsymbol{y})].$$

In order to utilise the SMM, it is necessary to have a fully specified stochastic model from which we can obtain a series of simulated time series $\boldsymbol{y^{\theta}}$ of length $T_{sim} \geq T_{emp}$. Let us consider a model $f(\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is the vector of parameters of the particular model. Relying on a single time series to approximate the moment vector is prone to randomness, making further optimization difficult. To address this, it is useful to simulate multiple runs from the model, enabling us to construct a $T_{sim} \times N_M$ matrix, where N_M is a number of simulated time series:

$$\boldsymbol{Y}^{\boldsymbol{\theta}} = [\boldsymbol{y}_1^{\boldsymbol{\theta}}, \dots, \boldsymbol{y}_{N_M}^{\boldsymbol{\theta}}].$$

To determine moments from the simulated data matrix, we compute the average of the sample moments for each time series:

$$\boldsymbol{m}^{sim}(\boldsymbol{\theta}) = [m_1(\boldsymbol{Y}^{\boldsymbol{\theta}}), \dots, m_D(\boldsymbol{Y}^{\boldsymbol{\theta}})]$$
$$m_d(\boldsymbol{Y}^{\boldsymbol{\theta}}) = \frac{1}{N_M}\sum_{n=1}^{N_M} m_d(\boldsymbol{y}_n^{\boldsymbol{\theta}}),\ d = 1, \dots, D,$$

which provides us with a vector of simulated moments, similar to what we have in the empirical case.

As previously noted, parameters are optimized by minimizing the weighted distance between empirical and simulated time series moments. We first define the difference between moments:

$$h(\boldsymbol{m}^{emp}, \boldsymbol{m}^{sim}(\boldsymbol{\theta})) = \boldsymbol{m}^{emp} - \boldsymbol{m}^{sim}(\boldsymbol{\theta}).$$

The loss function is then defined as:

$$J(\boldsymbol{\theta}) = h(\boldsymbol{m}^{emp}, \boldsymbol{m}^{sim}(\boldsymbol{\theta}))' \boldsymbol{W} h(\boldsymbol{m}^{emp}, \boldsymbol{m}^{sim}(\boldsymbol{\theta})),$$

where $\boldsymbol{W} \in R^{D\times D}$ is a positive definite weighting matrix. Then, we can define the SMM estimator as the minimization of the loss function:

$$\hat{\boldsymbol{\theta}}_{SMM} = \arg\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}),$$

where $\boldsymbol{\theta} \in \Theta$, Θ is the parameter space. In cases when we use more moments than we estimate parameters, then it is convenient to use J-test of overidentifying restrictions to evaluate the joint compatibility of all D moments:

$$\bar{f} = J(\hat{\boldsymbol{\theta}}) \xrightarrow{T\to\infty} \chi^2_{D-K}.$$

The null hypothesis $H_0 : \bar{f} = 0$ indicates that the model with the given set of moments sufficiently approximates the data-generating process. Rejecting the null hypothesis suggests that, in at least one dimension, the model fails to replicate the distributional characteristics of empirical data.

3.1.2 Weighting Matrix

The weighting matrix emphasizes stable moments more than those with higher variability. The distance between empirical and simulated moments should be more influential on moments we can estimate more precisely than those with higher variability. It is crucial to note that individual moments are not independent but can be highly correlated, potentially affecting the optimization process. Therefore, it is important to consider adding this information to the weighting matrix. From all stated above, it would be natural to calculate the weighting matrix $\boldsymbol{W}$ from the covariance matrix $\boldsymbol{\Sigma}$ of moments:

$$\boldsymbol{W} = \boldsymbol{\Sigma}^{-1}.$$

One approach to obtaining the covariance matrix is to use the Newey-West method. As shown by Franke (2009) and Chen & Lux (2018), the Newey-West approach is efficient and consistent even with heteroskedasticity and autocorrelation. Therefore, it provides the most efficient SMM estimator. However, this efficiency does not necessarily translate to small samples, as shown by Altonji & Segal (1996); it results in a biased GMM estimator and even the identity matrix can outperform it.

To overcome this limitation and address finite-sample issues, Franke & Westerhoff (2012) suggest estimating the covariance matrix with a bootstrap method. A block bootstrap approach is employed to calculate the covariances for individual moments. Specifically, non-overlapping blocks of 250 observations are used for short-memory moments (mean, variance, lower-order autocorrelations) and 750 observations for long-memory moments (higher-order autocorrelations).

In this study, we follow the approach outlined by Zila & Kukacka (2023), where the block bootstrap is improved by incorporating overlapping blocks to reduce small sample bias. The non-overlapping approach of sizes up to 750 observations results in small blocks and can suffer from small sample bias, similar to the Newey-West approach. Introducing overlapping blocks significantly increases the number of blocks and mitigates the impact of small sample bias.

Let us obtain the moment vector from the sample block $\boldsymbol{m}^b$, where $b = 1, \ldots, B$ and B is the total number of bootstrapped samples. Then, the estimate of the covariance matrix is given by:

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{B}\sum_{b=1}^{B}(\boldsymbol{m}^b - \overline{\boldsymbol{m}})(\boldsymbol{m}^b - \overline{\boldsymbol{m}})',$$

where $\overline{\boldsymbol{m}} = \frac{1}{B}\sum_{b=1}^{B}\boldsymbol{m}^b$. Then the weighting matrix $\boldsymbol{W}$ is obtained by taking the inverse of the estimated covariance matrix $\hat{\boldsymbol{\Sigma}}$ from the bootstrap procedure.

3.2 Non-parametric Simulated Maximum Likelihood

Kristensen & Shin (2012) introduce the NPSML and present it in the context of dynamic models. However, its universality can be applied to almost any type of model. Lee & Ingram (1991) demonstrate NPSML on discrete choice models without additive errors. The properties of the method in the context of univariate financial ABMs are investigated in Kukacka & Barunik (2017), and later, the method is extended to multivariate macroeconomic ABMs by Kukacka & Sacht (2023).

The primary advantage of this method is its general applicability and the fact that it does not necessarily need a stationary process. The usage of NPSML is mainly in models where the conditional likelihood is intractable, as the method is computationally demanding. The fundamental idea is to take the model for a given parameter set and use the independent variables for a particular data point to simulate the dependent variable N_L times. Utilizing KDE, we obtain the distribution of the conditional likelihood for a specified data point under the model with the given parameter set. Subsequently, the likelihood for all data points is computed, providing the likelihood of the data under the given model. The optimization process then seeks to find the parameter set with the best simulated likelihood.

3.2.1 Formal Definition

In the formal definition, we follow the approach outlined by Kristensen & Shin (2012) and Kukacka & Barunik (2017). Consider a data process of T observations, $\{(y_t, \boldsymbol{x}_t)\}_{t=1}^{T}$, where $y_t \in \mathbb{R}^k$ and $\boldsymbol{x}_t \in \chi_t$. The space χ_t may change over time, and the process is permitted to be nonstationary. The vector $\boldsymbol{x}_t$ may include both exogenous explanatory variables and lagged dependent variables

y_t. Additionally, let's assume that a fully parametric model is responsible for generating the data process:

$$y_t = f_t(\boldsymbol{x}_t, \epsilon_t; \boldsymbol{\theta}),\ t = 1, \ldots, T,$$

where $\boldsymbol{\theta} \in \Theta$ represents the unknown parameter vector to be optimized, and ϵ_t is an i.i.d. noise term with a known distribution F_ϵ. Without loss of generality, we can assume that ϵ_t is independent of both t and $\boldsymbol{\theta}$.

Let's also consider a conditional density related to the model in question:

$$P(y_t \in A | \boldsymbol{x}_t = \boldsymbol{x}) = \int_A p_t(y|\boldsymbol{x}; \boldsymbol{\theta})\, \mathrm{d}y,\ t = 1, \ldots, T,$$

for any Borel set $A \subseteq \mathbb{R}^k$, a standard approach for estimating $\boldsymbol{\theta}$ is to maximize the conditional log-likelihood:

$$\bar{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}\in\Theta} L_T(\boldsymbol{\theta}), \quad L_T(\boldsymbol{\theta}) = \sum_{t=1}^{T} \log p_t(y_t|\boldsymbol{x}_t; \boldsymbol{\theta}).$$

When the conditional density $p_t(y_t|\boldsymbol{x}_t; \boldsymbol{\theta})$ lacks a closed-form solution, making it impossible to derive the exact likelihood function, the maximum likelihood estimator becomes infeasible. This occurs when the inverse of $f_t(\boldsymbol{x}_t, \epsilon_t; \boldsymbol{\theta})$ either does not exist or does not have a closed-form expression. A common problem that arises in complex ABMs.

However, let's assume that we are still able to simulate observations from the model. In that case, the presented NPSML would then yield a simulated version of the conditional density. For all $1 \leq t \leq T, y_t \in \mathbb{R}^k, \boldsymbol{x_t} \in \chi_t$, and $\boldsymbol{\theta} \in \Theta$, we want to derive the simulated version of $p_t(y_t|\boldsymbol{x_t}; \boldsymbol{\theta})$. Utilizing the knowledge of F_ϵ, we produce N_L i.i.d. samples $\{\epsilon_i\}_{i=1}^{N_L}$ to calculate:

$$y_{t,i}^{\boldsymbol{\theta}} = f_t(\boldsymbol{x_t}, \epsilon_i; \boldsymbol{\theta}), \quad i = 1, \ldots, N_L.$$

These N_L simulated random variables $\left\{y_{t,i}^{\boldsymbol{\theta}}\right\}_{i=1}^{N_L}$ are i.i.d., and by design, they follow the target distribution $y_{t,i}^{\boldsymbol{\theta}} \sim p_t(\cdot|\boldsymbol{x}_t; \boldsymbol{\theta})$. By employing kernel methods, we can approximate $p_t(y_t|\boldsymbol{x}_t; \boldsymbol{\theta})$. Let us define:

$$\hat{p}_t(y_t|\boldsymbol{x}_t; \boldsymbol{\theta}) = \frac{1}{N_L} \sum_{i=1}^{N_L} K_h(y_{t,i}^{\boldsymbol{\theta}} - y_t).$$

We define the kernel $K_h(v)$ as $K_h(v) = K(v/h)/h^k$, where $K : \mathbb{R}^k \mapsto \mathbb{R}$ and

$h > 0$ represents the bandwidth. According to Kristensen & Shin (2012), given the regularity conditions for p_t and K, we obtain the following results:

$$\hat{p}_t(y_t|\boldsymbol{x}_t;\boldsymbol{\theta}) = p_t(y_t|\boldsymbol{x}_t;\boldsymbol{\theta}) + O_p(1/\sqrt{N_L h^k}) + O_p(h^2), \quad N_L \to \infty,$$

when $h \to 0$ and $N_L h^k \to \infty$, then the remainder terms are $o_p(1)$. To clarify notation, O_p means a term grows at a certain rate with high probability, while o_p means a term becomes negligible compared to another as the sample size gets very large.

After acquiring the simulated conditional density, we can then compute the simulated MLE of $\boldsymbol{\theta}$:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}\in\Theta} \hat{L}_T(\boldsymbol{\theta}), \quad \hat{L}_T(\boldsymbol{\theta}) = \sum_{t=1}^{T} \log \hat{p}_t(y_t|\boldsymbol{x}_t;\boldsymbol{\theta}).$$

In numerical optimization, we use the same set of draws for every value of $\boldsymbol{\theta}$. Furthermore, we can apply the same batch of draws across different values of t. For numerical optimization to be feasible, $\hat{L}_T(\boldsymbol{\theta})$ needs to be continuous and differentiable in $\boldsymbol{\theta}$. This is valid when K and $\boldsymbol{\theta} \mapsto f_t(\boldsymbol{x}_t, \epsilon_t; \boldsymbol{\theta})$ are continuously differentiable with respect to $r \geq 0$.

3.2.2 Properties

The broad applicability, such as the observations y_t not needing to be i.i.d. or stationary, is due to the density estimator being based on i.i.d. samples that are independent of the structure present in the observed data. Kernel estimators often face the curse of dimensionality; to manage the variance in high-dimensional models, a substantial number of simulations are required. However, in NPSML, extra smoothing is applied beyond the kernel estimation. The summation of individual likelihoods acts as an additional smoothing mechanism, managing variance and maintaining the standard parametric rate of $1/N_L$.

However, the drawback is that for a fixed N_L and $h > 0$, the NPSML acts as a biased estimator of the log-likelihood function. To achieve consistency, we need to let $N_L \to \infty$ and $h \to 0$. Selecting the appropriate bandwidth size h is crucial for a given sample and simulation size. However, if we can make more robust assumptions about the model's identification, the choice of bandwidth becomes less critical. For instance, the standard identification assumption in the stationary case is:

$$\mathbb{E}[\log p(y_t|\boldsymbol{x}_t;\boldsymbol{\theta})] < \mathbb{E}[\log p(y_t|\boldsymbol{x}_t;\boldsymbol{\theta}_0)] \quad \forall\boldsymbol{\theta} \neq \boldsymbol{\theta}_0.$$

According to Altissimo & Mele (2009) and Kristensen & Shin (2012), if stronger identification assumptions are met, it is possible to demonstrate that the estimator remains consistent for any fixed $0 < h \leq \bar{h}$, for some $\bar{h} > 0$ as $N_L \to \infty$. It is important to note that with a fixed $h > 0$, the estimator is not fully efficient, and the choice of h remains. From a theoretical perspective, it guarantees that the estimator can accurately identify the parameters in large finite samples. This indicates that the methodology is quite robust to the choice of h (Kristensen & Shin 2012).

Given a suitably selected sequence $N_L = N_L(T)$ and $h = h(N_L)$, if the theoretical convergence of the simulated conditional density to the true density is achieved, we would expect that $\hat{\boldsymbol{\theta}}_{NPSML}$ and $\tilde{\boldsymbol{\theta}}_{ML}$ exhibit the same asymptotic properties. Indeed, Kristensen & Shin (2012) establish that, under a broad set of conditions, $\hat{\boldsymbol{\theta}}_{NPSML}$ is first-order asymptotically equivalent to $\tilde{\boldsymbol{\theta}}_{ML}$. This equivalence holds even in the presence of nonstationarity and when the response variable has a mixed distribution, combining both continuous and discrete components. To establish higher-order asymptotic properties of $\hat{\boldsymbol{\theta}}_{NPSML}$, additional assumptions, such as stationarity, are needed.

To ensure that $\hat{p} \to p$ converges sufficiently quickly for the asymptotic equivalence of $\hat{\boldsymbol{\theta}}_{NPSML}$ and $\tilde{\boldsymbol{\theta}}_{ML}$, it is necessary to verify a set of general conditions that are typically satisfied by most models. For the model and its corresponding conditional density, Kristensen & Shin (2012) establishes a set of regularity conditions that meet these general requirements. These conditions are satisfied under uniform rates of convergence for kernel estimators (Kristensen 2009). Subsequently, the kernel K should be selected from a broad family of high-order bias or bias-reducing kernels. Furthermore, to satisfy that the $\boldsymbol{\theta}_{NPSML}$ behaves well asymptotically on the log-likelihood function and the associated MLE, we impose the standard conditions for the consistency of MLE in stationary and ergodic models.

3.2.3 Extension to Models with Latent Variables

Many ABMs involve latent variables, and it is essential to modify NPSML in order to be able to deal with latent dynamics. In this extension, we follow the approach outlined by Kristensen & Shin (2012) and Creel & Kristensen (2012). Suppose that y_t is generated from:

$$[y_t, w_t] = f(y_{t-1}, w_{t-1}, \epsilon_t; \boldsymbol{\theta}),$$

where $\epsilon_t \overset{i.i.d.}{\sim} F_\epsilon$ and w_t is an unobservable latent variable.

To compute the likelihood function, we need to determine the conditional density using all available past information:

$$p_t(y_t | y_{t-1}, y_{t-2}, \dots, y_0; \boldsymbol{\theta}).$$

The growing information set can become quite complex. Nevertheless, it is feasible to construct the Limited Information Likelihood (LIL), particularly its simulated variant. We define the simulated LIL as follows:

$$L_T(\boldsymbol{\theta}) = \sum_{t=1}^{T} \log p_t(y_t | \boldsymbol{x}_t; \boldsymbol{\theta}).$$

Where the set of conditioning variables $\boldsymbol{x}_t = (y_{t-1}, \dots, y_{t-m+1})$, $m \geq 2$ is chosen by the researcher and denotes the number of time-steps we want to simulate, the conditioning set is then of the size $m - 1$. It is possible to use $m = 1$, and then we are only simulating y_t for one step and calculating the LIL without the conditioning set. Therefore, it becomes almost identical to the standard NPSML.

LIL impose efficiency loss compared to the full likelihood when estimating $\boldsymbol{\theta}$, but it has a lower computational burden and is easier to implement. Let us start by simulating N_L trajectories of $\{y^{\boldsymbol{\theta}}_{j,i}\}_{j=t-m+1}^{t}$ for $i = 1, \dots, N_L$ by

$$[y^{\boldsymbol{\theta}}_{j,i}, w^{\boldsymbol{\theta}}_{j,i}] = f(y^{\boldsymbol{\theta}}_{j-1,i}, w^{\boldsymbol{\theta}}_{j-1,i}, \epsilon_{j,i}; \boldsymbol{\theta}), \quad j = t - m + 1, \dots, t,$$

where the collection of $\epsilon_{j,i}$ are i.i.d. draws from F_ϵ. Note that we assume that the initial values (y_{t-m}, w_{t-m}) are known. In practice, the w_{t-m} is unknown. However, we mitigate this problem by simulating the model N_L times for the specified burn-in period with given $\boldsymbol{\theta}$. We obtain a vector of $\{w_i\}_{i=1}^{N_L}$, which is then used as the initial value for the latent variable in the trajectory simulation. However, it brings another randomness into the estimation process, which leads to the need for a larger N_L.

Then, we can utilize a modified kernel estimator to approximate a simulated version of $p_t(y_t|\boldsymbol{x}_t;\boldsymbol{\theta})$:

$$\check{p}_t(y_t|\boldsymbol{x}_t;\boldsymbol{\theta}) = \frac{\sum_{i=1}^{N_L} K_h(y_{t,i}^{\boldsymbol{\theta}} - y_t) K_h(\boldsymbol{x}_{t,i}^{\boldsymbol{\theta}} - \boldsymbol{x}_t)}{\sum_{i=1}^{N_L} K_h(\boldsymbol{x}_{t,i}^{\boldsymbol{\theta}} - \boldsymbol{x}_t)},$$

where $\boldsymbol{x}_{t,i}^{\boldsymbol{\theta}} = (y_{t-1,i}^{\boldsymbol{\theta}}, \dots, y_{t-m+1,i}^{\boldsymbol{\theta}})$ is a conditioning set. A similar approach is taken by Altissimo & Mele (2009).

A few potential problems can make the convergence of $\check{p}$ slower relative to $\hat{p}$. Firstly, the dimension of $(y_{t,i}^{\boldsymbol{\theta}}, \boldsymbol{x}_{t,i}^{\boldsymbol{\theta}})$ can become overwhelming. Furthermore, there is now a dependent structure inside simulated variables. Moreover, the optimization procedure requires choosing a larger N_L to approximate conditional densities correctly. The assumption of stationarity would be helpful to deal with the dependent structure. However, Kristensen & Shin (2012) show that the NPSML in the LIL adjustment can also work under nonstationarity. Karlsen & Tjøstheim (2001) and Bandi & Phillips (2003) demonstrate that with a nonstationary recurrent Markov process, the kernel estimator is consistent and asymptotically mixed normally distributed, but the convergence is much slower and path-dependent.

3.3 Approximate Bayesian Computation

Rubin (1984) sets the grounding for the ABC algorithm for applied Bayesian inference, and Tavaré *et al.* (1997) demonstrate its first empirical application. ABC is a likelihood-free method that circumvents the likelihood computation using distance methods. Similar to SMM, it uses simulations of the process to approximate a simulated moment vector, which is then compared with the empirical moment vector. ABC can be considered the Bayesian counterpart of the SMM as the distance comparison part is the same. However, instead of optimizing to obtain a point estimate, ABC employs particle filters or other techniques to accept proposal particles whose simulated moment vector is close enough to the empirical moment vector. The accepted particles provide us with a posterior distribution $p(\boldsymbol{\theta}|\boldsymbol{y})$, where $\boldsymbol{\theta}$ is the parameter set and $\boldsymbol{y}$ is the evidence, our data.

The simplest case of the particle filter is a rejection filter (Rubin 1984; Tavaré *et al.* 1997). The rejection filter incrementally draws particles from the prior distribution and calculates the distance between the moments vector by simulations. If the distance is lower than a threshold set by the researcher, the particle is accepted. The problem with the rejection sampler is that the rejection rate is very large, imposing a high computational burden. Alternatives include MCMC (Marjoram *et al.* 2003), which creates a Markov chain by sequential steps of moves of particles to approximate the posterior distribution, where each step is either accepted or rejected. Secondly, one can utilize SMC (Drovandi & Pettitt 2011), which employs importance sampling to create a sequence of distributions, starting with the prior and ending with the desired posterior distribution. The individual distributions in SMC are formed from weighted particles sampled from the precedent distribution and resampled so that the sequence of distributions converges to the posterior distribution.

In our study, we follow the approach of Lux (2023a), where four variants of ABC are compared on the financial ABMs. The first is with a rejection sampler, and the remaining three use different variants of SMC. The rejection sampler proves to have the highest computational burden. Our study employs the third method, which utilizes an SMC variant from Drovandi & Pettitt (2011). This method, as shown in Lux (2023a), achieves the best results regarding precision and computational complexity. It uses the fact that, in each step of SMC, it drops τ share of particles, and the rest are resampled to keep the total number of particles constant. It automatically derives the acceptance threshold δ_s of the distance between simulated and empirical for the following step. Then, the algorithm employs an MCMC kernel of the invariant distribution to update the particles. The updates run R_s times for each particle, where R_s is derived from the acceptance rate from the last step to ensure that each particle is moved with a set probability threshold. The acceptance is based on whether the distance between the simulated moment vector using the proposed particle and the empirical moment vector is below the threshold. Lastly, when the final threshold δ_{S_A} is reached, the final particles are corrected with a linear adjustment that extrapolates the particles to the limiting case $\delta = 0$.

3.3.1 Formal Definition

We mainly follow Drovandi & Pettitt (2011) and Lux (2023a). ABC aims to get samples from the joint approximate posterior distribution

$$p(\boldsymbol{\theta}, \boldsymbol{y}^{\boldsymbol{\theta}} | \varrho(\boldsymbol{y}, \boldsymbol{y}^{\boldsymbol{\theta}}) \leq \delta_{S_A}) \propto p(\boldsymbol{y}^{\boldsymbol{\theta}} | \boldsymbol{\theta}) \pi(\boldsymbol{\theta}) 1_{\varrho(\boldsymbol{y}, \boldsymbol{y}^{\boldsymbol{\theta}}) \leq \delta_{S_A}},$$

where $\boldsymbol{\theta}$ is the parameter vector, $\pi(\boldsymbol{\theta})$ the prior distribution, ϱ is the distance metric between summary statistics of the empirical data $\boldsymbol{y}$, and the simulated data $\boldsymbol{y}^{\boldsymbol{\theta}}$ obtained from the model with parametrization of $\boldsymbol{\theta}$, typically the moments are used as summary statistics. The final tolerance is denoted by δ_{S_A}. Lastly, the indicator function 1 is one when the distance condition is satisfied; otherwise, it is zero. Marginalization over the simulated data gives us the approximate posterior for the parameter:

$$p(\boldsymbol{\theta}|\varrho(\boldsymbol{y},\boldsymbol{y}^{\boldsymbol{\theta}}) \leq \delta_{S_A}) \propto \int_{\boldsymbol{y}^{\boldsymbol{\theta}}} p(\boldsymbol{y}^{\boldsymbol{\theta}}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})1_{\varrho(\boldsymbol{y},\boldsymbol{y}^{\boldsymbol{\theta}})\leq\delta_{S_A}}\,\mathrm{d}\boldsymbol{y}^{\boldsymbol{\theta}}.$$

To converge to this approximation, we utilize the version of SMC without likelihoods, where for the non-increasing sequence of the tolerances, $\delta_1 \geq \delta_2 \geq \cdots \geq \delta_{S_A}$, $s = 1, \ldots, S_A$, we subsequently create a following sequence of distributions:

$$\pi_s(\boldsymbol{\theta},\boldsymbol{y}^{\boldsymbol{\theta}}|\varrho(\boldsymbol{y},\boldsymbol{y}^{\boldsymbol{\theta}}) \leq \delta_s) \propto p(\boldsymbol{y}^{\boldsymbol{\theta}}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})1_{\varrho(\boldsymbol{y},\boldsymbol{y}^{\boldsymbol{\theta}})\leq\delta_s}.$$

Each distribution is approximated using N_A weighted particles, $\{\boldsymbol{\theta}_s^i, W_s^i\}$, $i = 1, \ldots, N_A$. By $\pi_0(\boldsymbol{\theta})$, we denote the importance distribution for the first target, typically the prior distribution. The original variant from Sisson *et al.* (2007) utilizes a Markov transition kernel and a backward Markov kernel. However, these kernels are, in fact, chosen to be equal, which leads to a much simpler weighting formula. However, this arbitrary choice results in bias (Beaumont *et al.* 2009). Correction of bias and choice of more optimal backward kernel turn the algorithm into $O(N^2)$, meaning the time it takes to run the algorithm increases proportionally to the square of the number of elements N. Lux (2023a) adopts an alternative approach by Drovandi & Pettitt (2011) where the forward and backward kernels are exchanged with MCMC kernel, which removes the bias and returns the algorithm to an $O(N)$. Marjoram *et al.* (2003) verifies the theoretical validity of the MCMC kernel without likelihoods. Under a suitable kernel, the MCMC step becomes just a rejection method based on description summaries. The incremental weights are derived from importance sampling; the current target is divided by importance distribution (the previous distribution in SMC step).

Unfortunately, the drawback is the possibility of duplicated particles. Therefore, the MCMC kernel is repeated R_s times for each particle to mitigate this problem. Using the acceptance rate γ_{s-1}^{acc} from the last iteration, we can update the value of R_s dynamically

$$R_s = \frac{\log(c)}{\log(1 - \gamma_{s-1}^{acc})},$$

where c is a constant the user chooses, and $1 - c$ denotes the probability that the particle is moved at least once with the assumption of binary trials.

The choice of the sequence of thresholds $\delta_1, \ldots, \delta_{S_A}$ can be crucial for quick convergence, and it can increase the running time substantially. In Drovandi & Pettitt (2011), the sequence is derived adaptively so that the particles are sorted based on their distances at the end of the iteration. Then, 100τ percent of particles with the largest distance are dropped. Then, the threshold for the next iteration is chosen as the maximum distance from the remaining particles. This step is theoretically grounded on the incremental weights which, with the MCMC kernel, are given by

$$\tilde{w}_s^i = \frac{1_{\varrho(\boldsymbol{y}, \boldsymbol{y}^{\boldsymbol{\theta}^i_{s-1}}) \leq \delta_s}}{1_{\varrho(\boldsymbol{y}, \boldsymbol{y}^{\boldsymbol{\theta}^i_{s-1}}) \leq \delta_{s-1}}},$$

therefore $W_s^i \propto \tilde{w}_s^i W_{s-1}^i$, and it is clear that the weights are ones for the remaining particles and zero for the dropped particles.

To keep the size of particles constant, the particles are resampled from the remaining ones to fill the dropped particles, and as already said above, they are updated with the MCMC kernel. The algorithm finishes when the maximum distance of the population is below the final threshold δ_{S_A}. Note that the need to choose the sequence of thresholds is replaced with only tuning the τ and δ_{S_A} parameters. The pseudo-code for ABC by Drovandi & Pettitt (2011) is described in Algorithm 1.

Following Lux (2023a), we utilize the moments as the summary statistics. The summary statistics function is the moment function $\boldsymbol{m}(\boldsymbol{y})$ from Section 3.1. The distance measure is identical to the one in SMM:

$$\varrho(\boldsymbol{y}, \boldsymbol{y}^{\boldsymbol{\theta}}) = h(\boldsymbol{m}^{emp}, \boldsymbol{m}^{sim}(\boldsymbol{\theta}))' \boldsymbol{W} h(\boldsymbol{m}^{emp}, \boldsymbol{m}^{sim}(\boldsymbol{\theta})),$$

where $\boldsymbol{W}$ is derived using approach from Subsection 3.1.2. Note that similarly to the SMM, the moments for given parameter vector are derived using multiple simulations of the data, therefore, it would be more convenient to denote $\boldsymbol{y}^{\boldsymbol{\theta}}$ as $\boldsymbol{Y}^{\boldsymbol{\theta}}$, however, for simplicity and the fact that we are mainly describing the SMC algorithm within ABC we follow the original notation.

Algorithm 1 An ABC algorithm from Drovandi & Pettitt (2011)

```
Let $N_\tau$ be the integer part of $(1-\tau)N_A$
for $i$ in 1 to $N_A$ do
    repeat
        draw particle from prior $\boldsymbol{\theta}^i \sim \pi(\cdot)$
        simulate data from the model $\boldsymbol{y}^{\boldsymbol{\theta}^i} \sim f(\cdot|\boldsymbol{\theta}^i)$
        $\varrho^i = \varrho(\boldsymbol{y}, \boldsymbol{y}^{\boldsymbol{\theta}^i})$
    until $\varrho^i \leq \delta_0$
end for
sort particles by $\varrho^i$
set the max distance $\delta_{max} = \varrho^{N_A}$
while $\delta_{max} > \delta_{S_A}$ do
    drop $N_\tau$ particles with the largest $\varrho$
    set the next threshold $\delta = \varrho^{N_A - N_\tau}$
    for $j$ in $N_A - N_\tau + 1$ to $N_A$ do
        resample $\boldsymbol{\theta}^j$ from $\{\boldsymbol{\theta}^i\}_{i=1}^{N_A - N_\tau}$
        for $k$ in 1 to $R$ do
            update $\boldsymbol{\theta}^j$ using an MCMC kernel of invariant distribution
        end for
    end for
    calculate $R$ from the acceptance rate of the MCMC step
    sort particles by $\varrho^i$
    set the max distance $\delta_{max} = \varrho^{N_A}$
end while
```

It is important to note that ABC converges to the so-called partial posterior $p(\boldsymbol{\theta}|\boldsymbol{m}^{emp})$, which acts as an approximation to the true posterior $p(\boldsymbol{\theta}|\boldsymbol{y})$. The partial posterior, conditional on moments $\boldsymbol{m}^{emp}$, is equal to the true posterior only if the moments are sufficient statistics, which generally exists only for the exponential family of distributions. Fearnhead (2018) summarizes the results on asymptotic properties of ABC, more precisely, the posterior concentration around the true value with a increasing sample size of the empirical data $T \to \infty$ and the threshold level decreasing with the increasing sample size, $\delta_T \to 0$. Furthermore, the identifiability condition needs to be met by the limit of the summaries for the true parameter vector. In order for the ABC to converge to the partial posterior, if the summary functions obey a central limit theorem with rate $\sqrt{T}$, then the tolerance level δ_T should decline as

$\delta_T = o(1/\sqrt{T})$. Otherwise, there is no Gaussian limit, and the convergence to the partial posterior is not satisfied, i.e., $\delta_T\sqrt{T} \to c > 0$ (Frazier *et al.* 2018).

3.3.2 Regression Adjustment

Using an ABC, we obtain $p(\boldsymbol{\theta}|\varrho(\boldsymbol{y}, \boldsymbol{y}^{\boldsymbol{\theta}}) \leq \delta_{S_A})$ which serves as an approximation to the posterior. It would be practical to have an approximation in the case where $\delta_{S_A} \to 0$ with the increasing sample size as it would guarantee the concentration of the posterior at the exact partial posterior $p(\boldsymbol{\theta}|\boldsymbol{m}^{emp})$. However, the sample size is usually fixed in empirical studies, and we need to compromise with a case of $\delta_{S_A} > 0$.

The solution can be linear regression adjustment (Lux 2023a), which extrapolates the finite δ_{S_A} to the limiting case $\delta_{S_A} = 0$. When we assume a linear relationship between accepted particles and summaries, then for each element of the particle set, we can perform a series of regressions:

$$\theta^{i,j} = \alpha_j + \boldsymbol{\beta}_j'(\boldsymbol{m}^{sim}(\boldsymbol{\theta}^i) - \boldsymbol{m}^{emp}) + \eta_{i,j},$$

where $\theta^{i,j}$ denotes the j-th parameter of the i-th accepted particle for posterior $p(\boldsymbol{\theta}|\varrho(\boldsymbol{y}, \boldsymbol{y}^{\boldsymbol{\theta}}) \leq \delta_{S_A})$, $\boldsymbol{m}^{sim}(\boldsymbol{\theta}^i) - \boldsymbol{m}^{emp}$ is the difference between moment vectors of the simulated data and the empirical data for the given particle. Lastly, $\eta_{i,j}$ is the error term. Note that the regression is conducted separately for each parameter. To reduce the influence of outliers and increase the concentration around $\delta = 0$, we utilize the weighted regression, where the weights are calculated using the distance between moments to emphasize particles whose distance to the empirical moments is closer to 0. The weights are determined using a bounded kernel function K_δ: $\hat{w}_i \propto K_\delta[\varrho(\boldsymbol{y}, \boldsymbol{y}^{\boldsymbol{\theta}^i})]$; the thesis utilizes the Epanechnikov kernel as recommended by Blum (2017).

The regression coefficient $\hat{\boldsymbol{\beta}}_j$, $\hat{\alpha}_j$ and the residuals $\hat{\eta}_{i,j}$ are used for correcting the sample of particles:

$$\hat{\theta}^{i,j} = \theta^{i,j} - \hat{\boldsymbol{\beta}}_j'(\boldsymbol{m}^{sim}(\boldsymbol{\theta}^i) - \boldsymbol{m}^{emp}) = \hat{\alpha}_j + \hat{\eta}_{i,j}.$$

Corrected posterior should be closer to the case where $\delta \to 0$ and the obtained posterior better approximates the partial posterior $p(\boldsymbol{\theta}|\boldsymbol{m}^{emp})$. The adjustment is a standard choice in ecology and evolutionary genetics research where the ABC is used more often than in economics. It is also possible to utilize more flexible approaches such as the nonlinear heteroskedastic model, ridge regression, or neural networks (Blum 2017). We stick to the simplest method, linear regression.

3.4 Bayesian Estimation

We conclude the methodology with the description of the BE. It is important to note at the beginning that the ABC is also a BE technique, but we separate them, and by BE, we refer to the likelihood-based methods. The likelihood-based methods have been gaining popularity in the ABMs field. Grazzini *et al.* (2017), Platt (2020) and Lux (2022) utilize MCMC as the sampling scheme to obtain posterior distribution. Lux (2023b) and Zhang *et al.* (2023), on the other hand, exploit the usage of the SMC while argument that likelihood of ABMs is often multimodal. Therefore, the MCMC can get stuck inside the local node of the likelihood function, where the SMC is a much more robust alternative.

We imitate the implementation from Zhang *et al.* (2023) who are able to estimate parameters of the macroeconomic ABM by approximating the likelihood function using the non-parametric approach with KDE method in the same manner as the NPSML[1]. There is also a similarity with the ABC as for obtaining the samples from posterior, it utilizes the SMC algorithm. However, the version of SMC is different than in the ABC.

Again, the SMC estimates the posterior distribution by construction of a sequence of distribution utilizing the importance sampling. The first step is drawing the particles from the prior distribution. This initialization step is followed with recursion. The recursion aims to create an intermediate distribution, which bridges the distributions from the precedent and the next iteration. In the beginning, these distributions are close to the prior, but in the end, they should converge to the posterior distribution.

Each recursion consists of three steps to create this sequence of distributions. Starting with correction, it is reweighting the particles from the last stage to reflect the density change in the new iteration and evaluate the particle's quality based on likelihood obtained using the same approach as in Section 3.2. The second step is selection, where the particles are either resampled using systematic resampling, or they are kept as they are. The decision to resample or not is made adaptively based on effective sample size, which assesses how much the weights are distributed evenly. Therefore, the main purpose of resampling

[1] Actually, our implementation of these two methods is calling the same function for the likelihood calculation.

is to deal with particle degeneracy and eliminate particles with near-zero weight for the price of some particle duplication. The recursion ends with a mutation. The Markov transition kernel propagates forward particles to adjust them to the current intermediate distribution. Following Zhang *et al.* (2023), the mutation utilizes the Metropolis-Hastings algorithm, and the Markov transition kernel uses a simple random walk with Normal distribution, where the covariance is adaptively derived using particles from the last iteration and the acceptance rate. The selection and mutation are crucial for the accuracy of the SMC as they keep the sample diversity and the spread of weights even across the particles, which improves the accuracy of the importance sampling (Herbst & Schorfheide 2014).

3.4.1 Formal Definition

The likelihood approximation is identical to the covered in Section 3.2. The SMC inside Section 3.3 is easier than the one covered here in BE as the version in ABC is special in that weights are either one or zero, where the importance sampling is reduced to the acceptance of the particle based on distance. Following the definition by Zhang *et al.* (2023) and Herbst & Schorfheide (2014), the SMC aims to obtain the posterior density given by

$$p(\boldsymbol{\theta}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}).$$

We abbreviate the notation of the right-hand side with $\dot{p}(\boldsymbol{\theta}) = p(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$. The cornerstone of each SMC algorithm is the importance sampling, which aims to approximate $p(\boldsymbol{\theta}|\boldsymbol{y})$ with the help of different, tractable density $\dot{g}(\boldsymbol{\theta})$, which should be easy to sample from. Then, the importance sampling is derived from the identity

$$\mathbb{E}_{p(\cdot|\boldsymbol{y})}[h(\boldsymbol{\theta})] = \int h(\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{y})\,\mathrm{d}\boldsymbol{\theta} = \frac{1}{Z}\int h(\boldsymbol{\theta})\dot{w}(\boldsymbol{\theta})\dot{g}(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\theta},$$

where $\dot{w}(\boldsymbol{\theta}) = \dfrac{\dot{p}(\boldsymbol{\theta})}{\dot{g}(\boldsymbol{\theta})}$, Z is normalizing constant such that $p(\boldsymbol{\theta}|\boldsymbol{y}) = \dfrac{1}{Z}\dot{p}(\boldsymbol{\theta})$ and h is placeholder function to represent quantity which expected value we want to estimate, e.g., $h(\boldsymbol{\theta}) = \boldsymbol{\theta}$, then $\mathbb{E}_{p(\cdot|\boldsymbol{y})}[h(\boldsymbol{\theta})]$ is the mean of $\boldsymbol{\theta}$ under the distribution $p(\boldsymbol{\theta}|\boldsymbol{y})$. Then suppose that we have $\boldsymbol{\theta}^i \overset{\text{iid}}{\sim} \dot{g}(\boldsymbol{\theta})$, $i = 1, \ldots, N$ then we can utilize the weights to calculate weighted average

$$\bar{h} = \sum_{i=1}^{N} h(\boldsymbol{\theta}^i)\tilde{W}^i, \quad \text{where} \quad \tilde{W}^i = \frac{\dot{w}(\boldsymbol{\theta}^i)}{\sum_{j=1}^{N} \dot{w}(\boldsymbol{\theta}^j)}.$$

Under regularity conditions (Geweke 1989), the estimate converges almost surely to $\mathbb{E}_{p(\cdot|\boldsymbol{y})}[h(\boldsymbol{\theta})]$ as $N \to \infty$. The accuracy of the approximation depends on the closeness of $\dot{g}(\boldsymbol{\theta})$ to $\dot{p}(\boldsymbol{\theta})$. We want to have the weights as even as possible.

In practice, finding good density $\dot{g}(\boldsymbol{\theta})$, which would lead to efficient importance sampling, takes a lot of work. Therefore, SMC utilize importance sampling to construct a sequence of particle approximations to intermediate distributions, which serve as the bridge between the prior and posterior; we index them by n and define them as

$$\pi_s(\boldsymbol{\theta}) = \frac{p_s(\boldsymbol{\theta})}{Z_s} = \frac{[p(\boldsymbol{y}|\boldsymbol{\theta})]^{\Phi_s}\pi(\boldsymbol{\theta})}{\int [p(\boldsymbol{y}|\boldsymbol{\theta})]^{\Phi_s}\pi(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\theta}},$$

for $s = 1, \ldots, S_B$. The sequence is controlled by the tempering schedule $\{\Phi_s\}_{s=1}^{S_B}$, where $\Phi_1 = 0$ and $\Phi_{S_B} = 1$. Note that for $s = 1$ the initial distribution is equal to the prior $\pi_0(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta})$. Therefore, the algorithm's initialization is quite straightforward and consists of only drawing N_B particles from prior and defining the particle's weight $W_1^i = 1$, obtaining $\{\boldsymbol{\theta}_1^i, W_1^i\}$, $i = 1, \ldots, N_B$. However, as shown by Herbst & Schorfheide (2014), the tempering schedule is important. The linear schedule results in the likelihood of dominating the prior too quickly. Therefore, using a schedule where the distance is smaller in the early stages is a good idea. Both, Herbst & Schorfheide (2014) and Zhang *et al.* (2023) utilize following schedule with $\lambda > 1$:

$$\Phi_s = \left(\frac{s-1}{S_B - 1}\right)^{\lambda}.$$

Based on terminology from Herbst & Schorfheide (2014), the transformation of particles to the next distribution in sequence is called recursion. The first step is correction. All particles from the iteration $s - 1$ are reweighted by incremental weights used for the normalization of weights from past iterations:

$$\tilde{w}_s^i = [p(\boldsymbol{y}|\boldsymbol{\theta}_{s-1}^i)]^{\Phi_s - \Phi_{s-1}}, \quad \tilde{W}_s^i = \frac{\tilde{w}_s^i W_{s-1}^i}{\frac{1}{N_B}\sum_{j=1}^{N_B} \tilde{w}_s^j W_{s-1}^j}.$$

Algorithm 2 An SMC algorithm from Zhang *et al.* (2023)

```
define the tempering schedule $\{\Phi_s\}_{s=1}^{S_B}$
Initialization:
for $i$ in 1 to $N_B$ do
    draw particle from prior $\boldsymbol{\theta}^i \sim \pi(\cdot)$ and set $W_1^i = 1$
end for
Recursion:
for $s$ in 2 to $S_B$ do
    Correction: $\tilde{w}_s^i = [p(\boldsymbol{y}|\boldsymbol{\theta}_{s-1}^i)]^{\Phi_s-\Phi_{s-1}}, \tilde{W}_s^i = \dfrac{\tilde{w}_s^i W_{s-1}^i}{\frac{1}{N_B}\sum_{j=1}^{N_B}\tilde{w}_s^j W_{s-1}^j}$
    Selection:
    calculate effective sample size $ESS_s = N_B/(\frac{1}{N_B}\sum_{i=1}^{N_B}(\tilde{W}_s^i)^2)$
    if $ESS_s < N_B/2$ then
        resample the particles using systematic resampling from multinomial distribution $\{\boldsymbol{\theta}_{s-1}^i, \tilde{W}_s^i\}_{i=1}^{N_B}$ and set $W_s^i = 1, i = 1, \dots, N_B$
    else
        keep particles and $W_s^i = \tilde{W}_s^i, i = 1, \dots, N_B$
    end if
    Mutation: calculate $\hat{c}_s$ and $\tilde{\boldsymbol{\Sigma}}_s$
    for $i$ in 1 to $N_B$ do
        propose particle $\boldsymbol{\vartheta}_s^i|(\boldsymbol{\theta}_s^i, \tilde{\boldsymbol{\Sigma}}_s) \sim \mathcal{N}(\boldsymbol{\theta}_s^i, \hat{c}_s^2\tilde{\boldsymbol{\Sigma}}_s)$
        define acceptance probability $\xi(\boldsymbol{\vartheta}_s^i|\boldsymbol{\theta}_s^i) = \min\left\{1, \dfrac{[p(\boldsymbol{y}|\boldsymbol{\vartheta}_s^i)]^{\Phi_s}\pi(\boldsymbol{\vartheta}_s^i)}{[p(\boldsymbol{y}|\boldsymbol{\theta}_s^i)]^{\Phi_s}\pi(\boldsymbol{\theta}_s^i)}\right\}$
        if accept particle with probability $\xi(\boldsymbol{\vartheta}_s^i|\boldsymbol{\theta}_s^i)$ then
            $\boldsymbol{\theta}_s^i = \boldsymbol{\vartheta}_s^i$
        else
            keep particle $\boldsymbol{\theta}_s^i = \boldsymbol{\theta}_s^i$
        end if
    end for
end for
```

The second step is selection; firstly, we calculate the effective sample size $ESS_s = N_B/(\frac{1}{N_B}\sum_{i=1}^{N_B}(\tilde{W}_s^i)^2)$. It measures how the weights are spread equally; as Herbst & Schorfheide (2014) note, uneven weights lead to inaccurate particle approximation of the distribution and proposes adaptive version selection used by Zhang *et al.* (2023) as well. If $ESS_s < N_B/2$, resample the particles using systematic resampling by draws from a multinomial distribution $\{\boldsymbol{\theta}_{s-1}^i, \tilde{W}_s^i\}_{i=1}^{N_B}$

and set $W_s^i = 1$. Otherwise, keep particles and their weights, $\boldsymbol{\theta}_s^i = \boldsymbol{\theta}_{s-1}^i$, $W_s^i = \tilde{W}_s^i$. Herbst & Schorfheide (2014) demonstrate that the systematic resampling adds noise to the approximation, but the subsequent even weights reduce the variance in the correction step. Therefore, it is important to balance these two effects.

Finally, mutation updates the particles to move them toward the current distribution via the Metropolis-Hastings algorithm with a Markov transition kernel (Chopin 2004). Zhang *et al.* (2023) propagate particles using proposal $\boldsymbol{\vartheta}_s^i$ from normal distribution:

$$\boldsymbol{\vartheta}_s^i | (\boldsymbol{\theta}_s^i, \tilde{\boldsymbol{\Sigma}}_s) \sim \mathcal{N}(\boldsymbol{\theta}_s^i, \hat{c}_s^2 \tilde{\boldsymbol{\Sigma}}_s),$$

where $\tilde{\boldsymbol{\Sigma}}_s$ is importance approximation of variance based on $\{\boldsymbol{\theta}_{s-1}^i, \tilde{W}_s^i\}_{i=1}^{N_B}$, and $\hat{c}_s$ is derived from average acceptance rate γ_{s-1}^{acc} in past recursion and for $s > 2$ it follows

$$\hat{c}_s = \hat{c}_{s-1} \hat{f}(\gamma_{s-1}^{acc}), \quad \hat{f}(x) = 0.95 + 0.1 \frac{e^{16(x-0.25)}}{1 + e^{16(x-0.25)}}.$$

We accept the proposal and let $\boldsymbol{\theta}_s^i = \boldsymbol{\vartheta}_s^i$ with probability $\xi(\boldsymbol{\vartheta}_s^i | \boldsymbol{\theta}_s^i)$, and otherwise, we keep the particle unchanged; the probability is calculated as

$$\xi(\boldsymbol{\vartheta}_s^i | \boldsymbol{\theta}^i) = \min \left\{ 1, \frac{[p(\boldsymbol{y} | \boldsymbol{\vartheta}_s^i)]^{\Phi_s} \pi(\boldsymbol{\vartheta}_s^i)}{[p(\boldsymbol{y} | \boldsymbol{\theta}_s^i)]^{\Phi_s} \pi(\boldsymbol{\theta}_s^i)} \right\}.$$

Then the approximation of $\mathbb{E}_{f(\cdot|\boldsymbol{y})}[h(\boldsymbol{\theta})]$ is given by

$$\bar{h}_{s,N_B} = \frac{1}{N_B} \sum_{i=1}^{N} h(\boldsymbol{\theta}_s^i) W_s^i.$$

The recursion is iterated for S_B times; the main advantage is that the proposal distributions are constructed adaptively, where the $\pi_s(\boldsymbol{\theta})$ is a proposal distribution for $\pi_{s+1}(\boldsymbol{\theta})$ making it more robust to initial values than MCMC (Zhang *et al.* 2023). Note that with $\hat{f}(0.25) = 1$, the $\hat{c}_s$ ensures that the average acceptance rate is around 25%. The researcher needs to choose the N_B, S_B and λ ex ante. The pseudo-code is shown in Algorithm 2.

Herbst & Schorfheide (2014) provide the formal analysis of the non-adaptive version of this algorithm, and under regularity conditions, the SMC approximation satisfies the strong law of large numbers and central limit theorems for $N_B \to \infty$ with other parameters fixed. They provides three theorems that establish the properties of the three main steps: correction, selection, and mutation. That is important as it characterizes the trade-offs inside the design of SMC algorithms. The Herbst & Schorfheide (2014) also discuss the asymptotic properties of the adaptive version used in our thesis and shows that the adaptive version of the transition kernel does not affect the asymptotic variance. The convergence also satisfies the adaptive version of the selection step (Del Moral *et al.* 2006).

3.4.2 Computational Stability

The formal definition of the algorithm works with likelihoods. However, as the likelihood is either the product of likelihoods for each time-stamp or exp of the sum of log-likelihoods for each time-stamp, therefore, with increasing sample size T the likelihood is prone to underflow; in other words, the computer rounds the given number to zero as it becomes too small for the computer to represent it accurately. It has a simple solution. Only the correction step and acceptance probability inside the mutation need to be rewritten to work with log-likelihoods instead of likelihoods. These adjustments are inline with Naesseth *et al.* (2019). We start with acceptance probability:

$$\log[\xi(\boldsymbol{\vartheta}_s^i|\boldsymbol{\theta}_s^i)] = \log\left[\min\left\{1, \frac{[p(\boldsymbol{y}|\boldsymbol{\vartheta}_s^i)]^{\Phi_s}\pi(\boldsymbol{\vartheta}_s^i)}{[p(\boldsymbol{y}|\boldsymbol{\theta}_s^i)]^{\Phi_s}\pi(\boldsymbol{\theta}_s^i)}\right\}\right].$$

Moreover, log is a continuous strictly monotonic increasing function. Therefore, we can put the log inside the min function, and with the help of basic properties of the log function, it is easy to obtain the following formula

$$\log[\xi(\boldsymbol{\vartheta}_s^i|\boldsymbol{\theta}_s^i)] = \min\left\{0, \Phi_s\log[p(\boldsymbol{y}|\boldsymbol{\vartheta}_s^i)] + \log[\pi(\boldsymbol{\vartheta}_s^i)] - \Phi_s\log[p(\boldsymbol{y}|\boldsymbol{\theta}_s^i)] - \log[\pi(\boldsymbol{\theta}_s^i)]\right\}.$$

As we can see now, the calculation works only with log probabilities, making it much more stable. To transform the final log probability to the original probability is straightforward with the exp function.

Now, we move to the correction step. The incremental weights are simple:

$$\log(\tilde{w}_s^i) = \log\{[p(\boldsymbol{y}|\boldsymbol{\theta}_{s-1}^i)]^{\Phi_s-\Phi_{s-1}}\} = \log[p(\boldsymbol{y}|\boldsymbol{\theta}_{s-1}^i)](\Phi_s - \Phi_{s-1}).$$

Subsequently, also deriving the formula for weights normalization:

$$\begin{aligned}\log(\tilde{W}_s^i) &= \log\left(\frac{\tilde{w}_s^i W_{s-1}^i}{\frac{1}{N_B}\sum_{j=1}^{N_B}\tilde{w}_s^j W_{s-1}^j}\right)\\ &= \log(\tilde{w}_s^i) + \log(W_{s-1}^i) - \log(1/N_B) - \log\left(\sum_{j=1}^{N_B}\tilde{w}_s^j W_{s-1}^j\right).\end{aligned}$$

The only problem remains in $\log(\sum_{j=1}^{N_B}\tilde{w}_s^j W_{s-1}^j)$. It is important for stability to use a logarithm of incremental weights. We utilize the LogSumExp (LSE) function:

$$\text{LSE}(x_1, \ldots, x_n) = \log\left[\sum_{i=1}^{N}\exp(x_i)\right].$$

To provide computational stability robust to underflow and overflow, the LSE is using the following trick under the hood:

$$\begin{aligned}\text{LSE}(x_1, \ldots, x_n) &= \log\left(\sum_{i=1}^{N}[\exp(x^*)\exp(x_i - x^*)]\right)\\ &= \log\left(\exp(x^*)\sum_{i=1}^{N}\exp(x_i - x^*)\right)\\ &= x^* + \log\left(\sum_{i=1}^{N}\exp(x_i - x^*)\right),\end{aligned}$$

where $x^* = \max\{x_1, \ldots, x_n\}$. This provides us with a tool to rewrite the sum inside the normalization of weights to the stable formula:

$$\begin{aligned}\log\left(\sum_{j=1}^{N_B}\tilde{w}_s^j W_{s-1}^j\right) &= \log\left(\sum_{j=1}^{N_B}\exp[\log(\tilde{w}_s^j W_{s-1}^j)]\right)\\ &= \log\left(\sum_{j=1}^{N_B}\exp[\log(\tilde{w}_s^j) + \log(W_{s-1}^j)]\right)\\ &= \text{LSE}\left[\log(\tilde{w}_s^1) + \log(W_{s-1}^1), \ldots, \log(\tilde{w}_s^{N_B}) + \log(W_{s-1}^{N_B})\right].\end{aligned}$$

These three adjustments turn the SMC into a stable algorithm robust to underflow.

Chapter 4

Implemented Models

This chapter overviews the theoretical models employed to compare simulated estimation methods. Section 4.1 first introduces a simple AR model to demonstrate the functionality of our methods. Then, Section 4.2 moves to a more difficult example with the classical ARMA-GARCH, a popular model in financial economics. Both of these models are typically estimated using traditional econometric methods, which we use as a additional benchmark to our methods. Lastly, Section 4.3 presents the financial ABM proposed by Franke & Westerhoff (2012).

4.1 Autoregressive Model

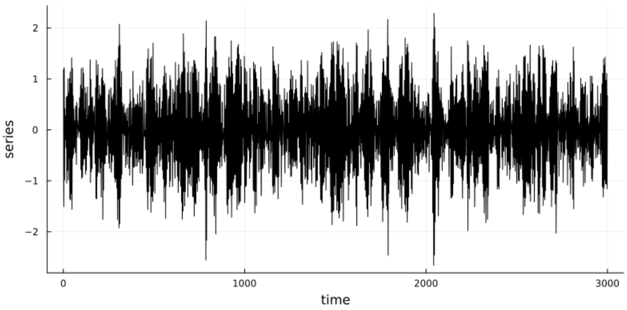
As the baseline model, we consider the AR model. This simple random process where the output variable linearly depends only on its past values and a stochastic term (noise). The AR(p) model is given by:

$$y_t = \sum_{i=1}^{p} \psi_i y_{t-i} + \epsilon_t,$$

where $\epsilon_t \sim N(0, \sigma^2)$. The AR($p$) is the mean-corrected process, and thus weak-sense stationary, if the polynomial $\psi(z) = 1 - \psi_1 z - \cdots - \psi_p z^p$ has all roots lying outside of the unit circle (Sathe & Upadhye 2022).

The AR process provides trivial dynamics, and it is used as the benchmark where all the methods stated above should perform well, thus demonstrating their validity. In the same manner, it is used in the ABM calibration methods comparison by Platt (2020), where the AR of order $p = 1$ is utilized. We add lag and use AR(2) as our testing model. The calibration follows a bootstrap simulation study by Souza & Neto (1996), therefore, we set the parameters in the following way: $\psi_1 = 0.2$, $\psi_2 = -0.9$, and $\sigma^2 = 0.1$. We estimate only ψ_1 and ψ_2, due to the theoretical properties of NPSML where the noise is assumed to be known a priori to the estimation. Therefore, the noise's variance parameter σ^2 is fixed. One simulation of the given specification of the AR(2) process is shown in Figure 4.1.

Figure 4.1: Simulated data from AR(2) model



Note: A time series of 3000 periods and a burn-in period 500.

4.2 ARMA-GARCH Model

Our second benchmark combines two traditional econometrics univariate models. The GARCH is utilized for the second moment modeling, and the mean equation is represented by ARMA process.

We start with the depiction of the GARCH model, introduced by Bollerslev (1986) to address the modeling of the second moment in time-series with the assumption of heteroscedasticity. The GARCH models the variance as an autoregressive moving average process, where there is a linear dependence between current variance and constant long-term variance, past variances, and past squared residuals. Formally, the GARCH(k,l) is given by three equations:

$$y_t = \nu_t + \epsilon_t, \quad \epsilon_t = \sigma_t \zeta_t$$

$$\sigma_t^2 = \omega + \sum_{i=1}^{k} \alpha_i \epsilon_{t-i}^2 + \sum_{j=1}^{l} \beta_j \sigma_{t-j}^2,$$

where dependent variable y_t is derived from the mean ν_t and the error term ϵ_t. The error term is deconstructed into normally distributed white noise innovation ζ_t and the conditional volatility σ_t. An autoregressive moving average process models the conditional volatility, where ω depicts the long-term baseline level of volatility, α represents the linear relationship of past squared errors on current volatility, and β measures the impact of the past conditional variances.

The magnitude of α parameters characterizes the persistence of the short-term shocks inside the model, and the persistence of the long-term shocks corresponds to the sum of both α and β (Campbell *et al.* 1998). The conditional variance must exist and be finite to ensure that the GARCH process is stationary. Therefore, the baseline level of volatility needs to be positive $\omega > 0$ and the rest of the parameters nonnegative $\alpha_i \geq 0$ for $i = 1, \dots, k$, and $\beta_j \geq 0$ for $j = 1, \dots, l$. For the conditional variance to remain finite, the sum of parameters must satisfy $\sum_{i=1}^{k} \alpha_i + \sum_{j=1}^{l} \beta_j < 1$ (Hull 2012). It is common only to use GARCH(1, 1) as it is already capable of capturing complex volatility behavior, and this simple specification can outperform more complicated methods (Miah *et al.* 2016).

As already stated, we utilize two processes, and we model the mean ν_t with ARMA, which is a combination of AR from Section 4.1 and Moving Average (MA). The ARMA(p, q) is given by

$$y_t = \nu + \sum_{i=1}^{p} a_i y_{t-i} + \sum_{j=1}^{q} b_j \epsilon_{t-j} + \epsilon_t,$$

where the ν denotes the long-term mean, a_i and b_i depict the linear dependence on past lags of the dependent variable and past error terms, respectively, errors ϵ_t are obtained using above specified GARCH process. The MA process is always stationary. Thus, the stationarity of the ARMA model is determined by the AR part; therefore, the ARMA must satisfy the same condition as already stated in Section 4.1.

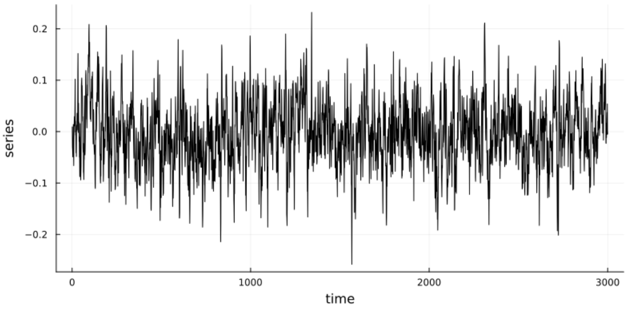
We chose simple ARMA(1,1) for our analysis. Therefore, the final model ARMA(1,1)-GARCH(1,1) used as our benchmark is given by:

$$\begin{aligned} y_t &= \nu + a_1 y_{t-1} + b_1 \epsilon_{t-1} + \epsilon_t \\ \epsilon_t &= \sigma_t \zeta_t \\ \sigma_t^2 &= \omega + \alpha_1 \epsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2. \end{aligned}$$

The σ_t can be considered as the latent variable. Therefore, this model provides us with the first benchmark in an environment with latent variables.

Similar benchmark is used in Platt (2020), where the author utilizes ARMA(2,2)-GARCH(2,0). The parameterization of the model is taken from an estimation study by Sathe & Upadhye (2022), where the parameters of the ARMA part are set in the following way: $\nu = 0$, $a_1 = 0.7$, and $b_1 = 0.1$. The parameters of the GARCH part are: $\omega = 0.001$, $\alpha_1 = 0.1$, and $\beta_1 = 0.3$. The single realization of the model with the given specification is demonstrated in Figure 4.2.

Figure 4.2: Simulated data from ARMA(1,1)-GARCH(1,1) model



Note: A time series of 3000 periods and a burn-in period 500.

4.3 Franke and Westerhoff (2012) Model

Finally, we present our main benchmark model, one of the most popular financial ABM. The roots of this model can be traced to the Franke (2008), where the individual agents are based on transition probabilities switching between two sentiments, optimism and pessimism. Franke & Westerhoff (2011) then introduce two groups of agents, chartist, and fundamentalist. Fundamentalists are considered the ones with long time horizons and base their demand on the deviation between the current price and the fundamental value. Chartists, on the other hand, analyze the market based on the latest price trends. A key feature of their model is that agents have the flexibility to alternate between strategies, making decisions based on recent price levels and a herding effect.

The model is further enriched by Franke & Westerhoff (2012), who introduce two potential switching mechanisms and outlines various methods for calculating the attractiveness of different strategies. In their work, Franke & Westerhoff (2012) propose seven distinct variations of the model and evaluate them based on how well they capture the typical characteristics of financial data. The model gains popularity and is used in many estimation studies. To name a few, Zila & Kukacka (2023) utilize the model to demonstrate the machine learning extension of the SMM. Lux (2022) and Platt (2020) use it as a benchmark in their study of the Bayesian estimation in the ABM field. Franke & Westerhoff (2016) then take the winner from the model variants and further studies its properties in the phase plane of the price and a majority index, which quantifies the share of given strategies. Our description only focuses on this variant of the model.

As previously mentioned, the model dynamics are influenced by two types of agents: chartists and fundamentalists. Through their interactions, a market maker determines the price for each period. The market maker adjusts the price according to the excess demand and supply from these two groups, so that the log price p_t at the given period is equal to

$$p_t = p_{t-1} + \mu(n^f_{t-1} d^f_{t-1} + n^c_{t-1} d^c_{t-1}),$$

where n^f_{t-1}, $n^c_{t-1} = 1 - n^f_{t-1}$ denote the proportions of fundamentalist and chartist agents in the market, respectively. The excess demand from each group is represented by d^f_{t-1} and d^c_{t-1}. The market marker responds to these imbalances by adjusting the price with a rate $\mu > 0$.

The demand of an average trader in each group is governed by a straightforward deterministic rule. Fundamentalists set their demand based on an inverse relationship with the difference between the price and its fundamental value $(p^* - p_t)$, where p^* is a log of the fundamental value, which is treated as an exogenous constant. As noted by Franke & Westerhoff (2012), the required stylized facts are obtained even without the random walk behavior of the fundamental value. On the other hand, the demand of chartists is determined by the recent price movement $(p_t - p_{t-1})$. The demands are then derived as

$$\begin{aligned} d^f_t &= \phi(p^* - p_t) + \epsilon^f_t,\ \epsilon^f_t \sim \mathcal{N}(0, \sigma^2_f) \\ d^c_t &= \chi(p_t - p_{t-1}) + \epsilon^c_t,\ \epsilon^c_t \sim \mathcal{N}(0, \sigma^2_c), \end{aligned}$$

nonnegative parameters ϕ and χ denotes the adjustment constants of the demands, noise terms ϵ^f_t and ϵ^c_t are added to the demands, and σ_f and σ_c are the volatilities of the mentioned noise terms. Independent noise terms for each demand are a crucial part of the model as they represent a certain heterogeneity as then the variation of the log returns $r_t = p_t - p_{t-1}$ acts as the sum of two normal distributions dependent on the fluctuation of the fundamentalists and chartists, $\sigma_t^2 = \mu^2[(n_{t-1}^f)^2\sigma_f^2 + (n_{t-1}^c)^2\sigma_c^2]$. The change of variance with time is why the model can match the stylized facts of the returns (Franke & Westerhoff 2012).

A switching mechanism controls the adjustment of population ratios. The superior variant of Franke & Westerhoff (2012) utilize the Discrete Choice Approach (DCA) where the impact is directly on n_t^f:

$$n_t^f = \frac{1}{1 + \exp(-\beta^{\text{fw}} a_{t-1})}, \quad n_t^c = 1 - n_t^f,$$

a_{t-1} denotes the relative attractiveness of fundamentalism over chartism. It is the main force driving the switching mechanism, and β^{fw} is the intensity of choice influencing the magnitude of the effect of the a_{t-1}. Note that a_{t-1} serves as a latent variable inside the model.

There are various combinations of how to specify the relative attractiveness a_t. The dominant variant in Franke & Westerhoff (2012) comprises three parts. The first principle is called herding, which means that the attractiveness of one group rises with the number of participants it already has. The implementation of herding is straightforward in that the a_t change proportionally to the difference between shares of the fundamentalists and chartists $n_t^f - n_t^c$ and the magnitude of this effect is controlled by $\alpha_n \geq 0$.

Secondly, it is possible that the agents in question may have an a priori preference for one of the strategies that can be measured by the parameter α_0. When the parameter is positive (negative), the agents tend towards fundamentalism (chartism).

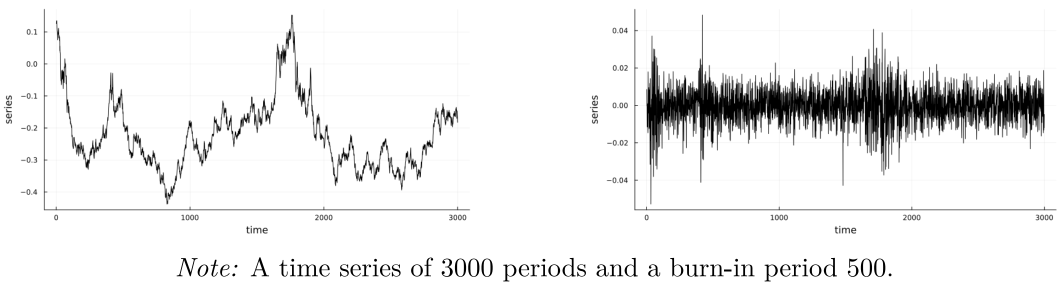
Finally, the third principle, price misalignment, indicates that chartism seems riskier as the deviation from the fundamental value increases. As a result, a_t grows in proportion to the squared deviation of the price from the fundamental value, $(p_t - p^*)^2$. The size of the influence on the relative attractiveness is measured by parameter $\alpha_p \geq 0$. Hence, the full equation is given by

$$a_t = \alpha_n(n_t^f - n_t^c) + \alpha_0 + \alpha_p(p_t - p^*)^2.$$

Our parametrization of the model follows Franke & Westerhoff (2012), where the model is calibrated so it can reasonably capture stylized facts of returns. The same parametrization is also used by Platt (2022) and Zila & Kukacka (2023). The parameters controlling the aggressiveness of the individual excess demands are set to $\phi = 0.12$ and $\chi = 1.5$. The factors determining the influence of the three principles on relative attractiveness are calibrated such that $\alpha_0 = -0.327$, $\alpha_n = 1.79$, and $\alpha_p = 18.43$. The remaining parameters are fixed during the estimation study and set so that $\sigma_f = 0.758$, $\sigma_c = 2.087$, $p^* = 0$, $\mu = 0.01$, and $\beta^{\text{fw}} = 1$. The volatility parameters of the noise terms σ_f and σ_c are fixed due to theoretical properties of the NPSML where the noise is assumed to be known a priori to the estimation. The fundamental value is redundant in a simulation study as treating it as a constant does not influence the model; nevertheless, in the empirical study, it needs to be calculated by the researcher (Franke & Westerhoff 2012; Bertschinger & Mozzhorin 2021). When you closely examine the Franke & Westerhoff (2012) model you conclude that the adjustment rate of the market maker μ and intensity of the choice β^{fw} are just multiplicative constants for ϕ, χ, σ_f, σ_c and α_0, α_n, α_p respectively, thus, should not be estimated (Franke & Westerhoff 2012; Bertschinger & Mozzhorin 2021). To verify this, we run a simple simulation exercise with multiple values of β^{fw}, and the estimation of β^{fw} always converges to 1, because the methods are scaling other parameters.

The single time series from simulation run with given parametrization is displayed in Figure 4.3, with the original price level in Figure 4.3a and the log returns $r_t = p_t - p_{t-1}$ in Figure 4.3b. In the simulation study, the methods with moments, ABC and SMM, utilize the log returns, and Simulated Maximum Likelihood (SML) methods, BE and NPSML, use the original price levels.

Figure 4.3: Simulated data from Franke and Westerhoff (2012) model


(a) Log price

(b) Log returns

Note: A time series of 3000 periods and a burn-in period 500.

Chapter 5

Simulation Setup

The chapter depicts the setting of the simulation exercises. Firstly, Section 5.1 describes the general framework used to compare the methods. Subsequently, Section 5.2 moves to the particular specification of the individual simulation estimation methods. Lastly, Section 5.3 is dedicated to the specific models, their optimization constraints and priors used by frequentist and Bayesian methods, respectively, and the selection of moments utilized by ABC and SMM.

5.1 General Setting

The study is conducted in `Julia 1.10.0-beta2`, which has native build-in parallel computing with `Distributed.jl` package. As the exercises are very computationally heavy, we take advantage of `MetaCentrum`, which provides the distributed computing infrastructure for researchers from the Czech Republic for free.

As the time-series length for all models, we have chosen $T_{emp} = 3000$. It is a sensible size corresponding to a dataset of over eight years of daily return, which is a realistic number to obtain in empirical financial studies. To eliminate the initial conditions, it is required to conduct a burn-in period where the initial simulated values are discarded, and after that, the T_{emp} number of data points are generated. Following Zila & Kukacka (2023) where the warm-up period is decided using the method from Welch (1983) for the Franke & Westerhoff (2012) model. The analysis shows that a 100 observation should be enough for the model's dynamics to settle; the authors then, for cautious reasons, use the burn-in period of size 200. We stick to this number for all our models.

The comparison of frequentist and Bayesian methods is not straightforward as their output is point estimate or posterior distribution, respectively. However, we start by simulating 100 time series, the so-called pseudo-empirical, under different random seeds. We would want a higher number than 100 to mitigate the randomness, but the computational resources constrain us. Then, the simulation exercise is conducted as follows: We take one pseudo-empirical time series. With Bayesian methods, BE and ABC, we estimate the posterior distribution with $N_A = N_B = 2000$ particles (this choice is more discussed in Section 5.2). In the case of the frequentist methods, SMM and NPSML, we iterate the method numeral times to obtain 96 point estimates. It is an important step, as the simulation methods are prone to randomness. Therefore, there is a need for a higher number of point estimates to average it away. The number 96 is chosen to align with the number of CPUs, as we primarily run simulations on 32 CPUs. Also, the time to obtain 96 point estimates roughly equals the posterior estimation between moment/likelihood estimation methods pairs. The final estimate for a given pseudo empirical time series is then a mean of 96 point estimates or a mean of 2000 particles, we call it mean estimate. The confidence or credible intervals are calculated as the quantiles of point estimates or particles, respectively. This is repeated for each time series; therefore, we end up with 100 estimates and 100 confidence/credible intervals for each method and model. The performance is then assessed using density plots of these 100 estimates and tables, which present the average of the estimates and the average of the confidence/credible intervals. In the case of the AR and ARMA-GARCH models, we can estimate them with traditional techniques. These techniques behave deterministically; therefore, each time series is estimated once, and we derive the standard confidence intervals, obtaining again 100 of these estimates and intervals in total. As noted by Kukacka & Zila (2024), the quantile version of the confidence/credible intervals within simulation-based methods acts as an approximation and should be named rather a confidence/credible intervals of the sample estimates.

The main analysis compares the quality of the estimates parameter by parameter. However, to have a single number to evaluate a given estimation method, we utilize the metrics used in Zila & Kukacka (2023)

$$RMSE_{\hat{\boldsymbol{\theta}}} = \frac{1}{K} \sum_{k=1}^{K} \sqrt{\frac{(\hat{\theta}_k - \theta_k^{true})^2 + \hat{\sigma}_k^2}{|\theta_k^{true}|}},$$

where the nominator is sum of the bias term $(\hat{\theta}_k - \theta_k^{true})^2$ and the estimated variance of the estimates $\hat{\sigma}_k^2$, it is normalized by the absolute value of the true value of the parameter $|\theta_k^{true}|$ to equalize of the effect of parameters with different magnitudes. The θ_k is calculated as the average of the 100 final estimates, and the $\hat{\sigma}_k$ is their standard deviation.

5.2 Methods Setting

For SMM and NPSML to converge to the solution, it is required to select an optimization algorithm. We utilize the differential evolution optimizer from the `BlackBoxOptim.jl` package, it is the recommended default algorithm from the authors of the library as it ranks on top of a tested set of problems. Zila & Kukacka (2023) exploit the same algorithm. We also conduct a simple exercise to compare the performance with other optimization algorithms, more precisely particle swarm optimization and BFGS algorithm, from `Optim.jl` library. However, the `BlackBoxOptim.jl` delivers the best results. We follow Zila & Kukacka (2023) and restrict the algorithm to 4000 calls to the loss function. The lower number increase the variance of the estimator as it can have happened that it do not converge; on the other hand, a higher number do not provide better results on outweigh the increase in the computational time.

For the simulated moments, used by SMM and ABC, we set the length of the simulated series equal to the pseudo empirical ones $T_{sim} = T_{emp}$. It is also essential to decide the number of simulated time series N_M, which are used to approximate the moment vector. We perform a simple simulation exercise with Franke & Westerhoff (2012) model where the values of N_M varies from 50 to 1600. It showcases that lower numbers lead to an increase in variability in the final estimate; although the computational complexity rises linearly with N_M, the performance is diminishing. Therefore, we have chosen $N_M = 400$ as a good trade-off between performance and complexity. Finally, the number of bootstrapped samples for weighting matrix calculation is set in line with Zila & Kukacka (2023) such that $B = 5000$.

For likelihood calculation we follow the study by Kukacka & Sacht (2023), where the conditional density is approximated using Gaussian kernel and the bandwidth $h = \left(\frac{4}{3N}\right)^{1/5} \hat{\sigma}$, where $\hat{\sigma}$ is the standard deviation of the simulated values for the given conditional density approximation. The number of the simulations is set using a similar exercise to the simulated moment's methods; we decide on $N_L = 400$ for analogous reasons. For ARMA-GARCH and Franke & Westerhoff (2012) models, we need to decide the trajectory length m to deal with the latent variable. The simulation exercises on given models where m ranged from 1 to 20 show that the likelihood with $m > 1$ quickly becomes flat and biased. Therefore, we use the $m = 1$, which gives us reasonable and the best-performing results. However, it means that the algorithm from Subsection 3.2.3 becomes normal NPSML described in Subsection 3.2.1 with added vector of the latent variable from the separate burn-in period of the model with given parameter set. The burn-in period in the likelihood approximation is increased to 1000 to ensure the model dynamics stabilizes for all parameter combinations. Finally, we pre-simulate and fix the random noise before each estimation run as it smooths out the likelihood function (Zhang *et al.* 2023).

Lux (2023a) and Zhang *et al.* (2023) utilize particle population of size $N_A = N_B = 1000$; we increase this number to $N_A = N_B = 2000$ as it improves performance, and the particle duplication is lower. For the BE, we set the number of iterations to $S_B = 150$, which is set such that the number of calls to the likelihood function is similar to NPSML. The parameter controlling the distance between bridge distributions $\lambda = 3.4$ is set in line with Zhang *et al.* (2023). The setting of hyperparameters of ABC is inspired with Lux (2023a), where the survival rate between iterations τ and the parameter controlling the average acceptance of the particle moves c are set such that $\tau = 0.5$ and $c = 0.01$. The initial δ_0 is theoretically set to infinity as we accept every draw from prior. Setting a good δ_0 is complex; the initialization acts as a rejection sampler and can become overwhelmingly long, and in our exercises, it does not depend on the initial quality of the particles for the SMC to successfully converge. Lux (2023a) set the final δ_{S_A} to 0.2 percent quantile of the distances of the initial particle sample. This threshold has been too low in our exercises. Therefore, we utilize different approach. We monitor the number of calls to the loss function, and after each iteration, we calculate the total number of calls. If it is over 300000, we stop the algorithm. This value equals the number of calls to the loss function by the BE and is a sufficient number for the ABC to converge.

5.3 Model Constraints and Moments

This section provides an overview of the simulation setup for individual models. The optimization algorithm used for NPSML and SMM must restrict each parameter's search range. Also, the Bayesian methods, ABC and BE, require the prior distribution as the initial point, which affects the final posterior distribution. Finally, the ABC and SMM depend on the moment set selection. Therefore, we list all this information in subsequent subsections.

There is a lack of literature on the prior selection in the financial ABMs framework. Studies mostly stick to the uninformative uniform prior, to name a few Lux (2022; 2023a); Platt (2020; 2022). Following this, all models utilize uniform prior with bounds identical to the optimization constraints. The uniform priors bring the Bayesian estimation closer to the frequentist methods (Lux 2023a).

5.3.1 AR(2)

For the optimization constraint and uniform prior of the parameters of the AR(2) model, we assume $\psi_1, \psi_2 \in [-1, 1]$. The main point of this range of values is to ensure that most of the parameters draw results into stationary series. The parameters' summary is shown in Table 5.1.

The moment set for methods with simulated moments consists of variance of raw returns and i-th-order auto-correlations, $i \in \{1, 2, 3\}$, of raw returns. It proves to be a sufficient moment set in our experiments.

Table 5.1: Parameters of AR(2) model and their boundaries

	Parameter	**True value**	**Constraint / Prior**
ψ_1	first lag coefficient	0.2	[-1, 1]
ψ_2	second lag coefficient	-0.9	[-1, 1]
	Fixed parameter		
σ^2	variance of the noise term	0.1	

Note: The optimization constraints and the boundary for uniform priors are identical; therefore, they are stated in a single column for simplicity. The noise variance σ^2 is fixed and not subject to estimation.

5.3.2 ARMA(1,1)-GARCH(1,1)

The choice of constraints and uniform priors are motivated in the same way as in Subsection 5.3.1 to ensure that the draws mostly result in a stationary series. The ARMA parameters are chosen such that $a_1, b_1 \in [-1, 1]$, and the GARCH parameters $\alpha_1, \beta_1 \in [0, 1]$. We assume $\nu \in [-1, 1]$ and $\omega \in [0, 1]$, respectively, for the long-term mean of returns and baseline volatility. The summary is shown in Table 5.2.

For ABC and SMM, we consider a moment set of size 10. We calculate variance, kurtosis, and skewness from raw returns. Then, we exploit the mean of absolute returns and the first and second order of raw, absolute, and squared returns. We do not find any more moments that would improve the estimation.

Table 5.2: Parameters of ARMA(1,1)-GARCH(1,1) model and their boundaries

	Parameter	True value	Constraint / Prior
ν	returns long term mean	0	[-1, 1]
a_1	autoregressive first lag	0.7	[-1, 1]
b_1	moving average first lag	0.1	[-1, 1]
ω	baseline volatility	0.1	[0, 1]
α_1	squared errors first lag	0.3	[0, 1]
β_1	conditional variance first lag	0.001	[0, 1]

Note: The optimization constraints and the boundary for uniform priors are identical; therefore, they are stated in a single column for simplicity.

5.3.3 Franke and Westerhoff (2012)

The initial ranges for the optimization constraint and uniform prior of parameters are inspired by Platt (2022), but most are moved during preliminary simulation tryouts. The range for the predisposition parameter is kept $\alpha_0 \in [-1, 1]$. The parameters of adjustment rates of excess demands and herding are slightly changed to $\phi, \chi \in [0, 3]$, and $\alpha_n \in [0.5, 4]$, respectively. The parameter controlling the effect of price misalignment is occasionally overestimated, and the upper bound is insufficient. Therefore, we increase the range to $\alpha_p \in [5, 50]$. The summary is depicted in Table 5.3.

As the moments set, we utilize the study from Zila & Kukacka (2023) where they use a union of moments sets from Chen & Lux (2018) and Franke & Westerhoff (2012) with additional three moments. The selection consists of variance, kurtosis, and i-th-order auto-correlations, $i \in \{1, 2, 3\}$, of raw returns. Subsequently, the mean, Hill estimator of the right tail at 2.5% and 5%, and j-th-order auto-correlations, $j \in \{1, 5, 10, 15, 20, 25, 50, 100\}$ of absolute returns are calculated. Finally, from squared returns, we evaluate the k-th-order auto-correlations, $k \in \{1, 5, 10, 15, 20, 20\}$. Even when the study from Zila & Kukacka (2023) focus on selecting the optimal minimal set, and shows that even small moment set can lead to superior performance. Their finding is also that the accuracy of estimates is not significantly declining when the weighting is precisely estimated. Therefore, we utilize the total moment set in our exercises as the weighting matrix proves to be satisfactory in capturing the covariances between moments, and therefore, the total moment set can be more robust with different parametrization.

Table 5.3: Parameters of Franke and Westerhoff (2012) model and their boundaries

	Parameter	**True value**	**Constraint / Prior**
ϕ	adjustment rate of fundamentalists	0.12	[0, 3]
χ	adjustment rate of chartists	1.5	[0, 3]
α_0	predisposition	-0.327	[-1, 1]
α_n	herding	1.79	[0.5, 4]
α_p	price misalignment	18.43	[5, 50]
	Fixed parameter		
μ	market maker adjustment rate	0.01	
β^{fw}	intensity of choice	1	
σ_f	volatility of fundamentalists' noise	0.758	
σ_c	volatility of chartists' noise	2.087	
p^*	log of fundamental value	0	

Note: The optimization constraints and the boundary for uniform priors are identical; therefore, they are stated in a single column for simplicity. The parameters μ, β, σ_f, σ_c and p^* are fixed and not subject to estimation.

Chapter 6

Results

The chapter presents the comparison results of the methods from the simulation exercises. Section 6.1 starts with results for the AR(2) model. Subsequently, Section 6.2 moves to the first model with latent variable, ARMA(1,1)-GARCH(1,1). Lastly, Section 6.3 concludes the chapter with the primary model from the ABM family, Franke & Westerhoff (2012) model. Note that when referring to the simulation-based methods, the confidence/credible intervals of the sample estimates are abbreviated simply as confidence/credible intervals.

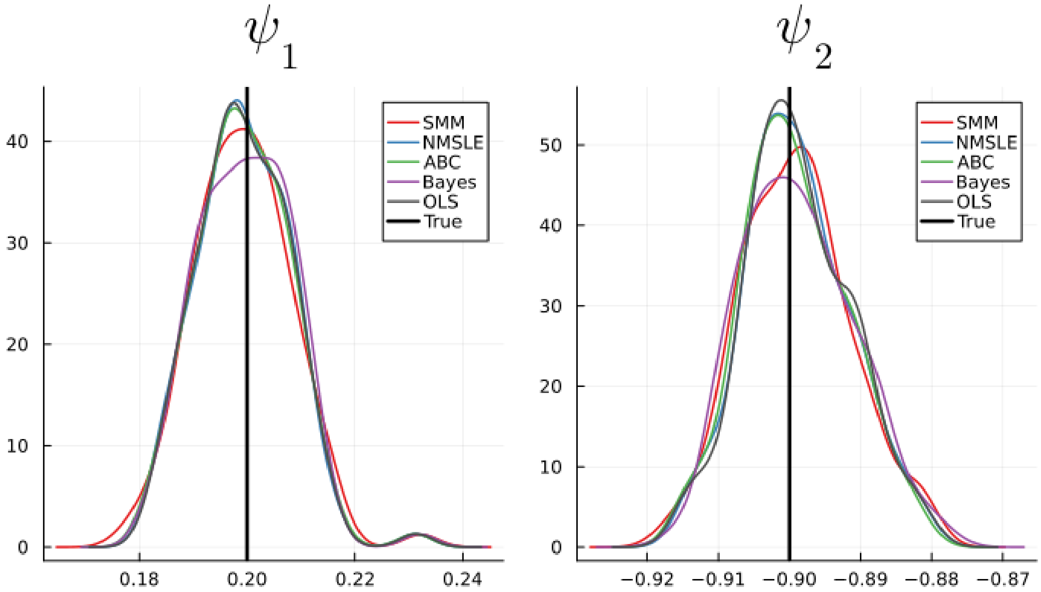
6.1 AR(2)

The AR(2) model serves as the baseline environment to test the functionality of our methods. The dynamics should be easy to capture; therefore, the purpose is to demonstrate that the methods are implemented correctly. The fact that the process can be estimated with classical econometrics methods provides an interesting comparison between simulation estimation methods and traditional Ordinary Least Squares (OLS).

The Figure 6.1 present the density plot of estimates of the parameters for each method, which are obtained as the mean of the replications of the SMM and NPSML or as the mean of the posterior distribution in case of ABC and BE from each of the 100 independent simulation runs. The densities are almost identical; there is no visible difference with the density of estimates from OLS, indicating that the simulation methods' performance can match the traditional method. However, there is a substantial difference in computational demands.

The more profound insight into the methods can be derived from the Table 6.1, where we present average estimates from the given simulation runs.

Figure 6.1: AR(2): Density of estimated parameters



Note: Kernel density plot of the mean estimates from each of the 100 independent simulations. `StatsModels.jl` and `GLM.jl` packages are used to obtain OLS estimates.

Table 6.1: AR(2): Simulation results using mean estimates

	θ_{true}	SMM	NPSML	ABC	BE	OLS
ψ_1	0.2	0.1993	0.1994	0.1994	0.1997	0.1994
		(0.1978, 0.2007)	(0.1947, 0.2042)	(0.1986, 0.2001)	(0.1834, 0.2158)	(0.1837, 0.2151)
ψ_2	-0.9	-0.8991	-0.8988	-0.8992	-0.8988	-0.8987
		(-0.9003, -0.8977)	(-0.9035, -0.8942)	(-0.8999, -0.8985)	(-0.9148, -0.8826)	(-0.9144, -0.8831)
$RMSE_{\hat{\theta}}$	-	0.0141	0.0133	0.0134	0.0139	0.0133

Note: The table presents average mean estimates of the AR(2) model's parameters over 100 independent simulations. The average 95% sample confidence/credible intervals of the sample estimates are inside () brackets. The ABC demonstrate estimates after linear adjustment. `StatsModels.jl` and `GLM.jl` packages are used to obtain OLS estimates and confidence intervals.

The individual estimates are similar across the methods, but more interesting are the 95% confidence and credible intervals of the sample estimates for the frequentist and the Bayesian methods, respectively. The broadest range is given by BE, indicating that the particles defining the posterior distribution are more spread than the frequentist's point estimates. On the other hand, the narrowest interval is given by ABC due to the post-processing of the final posterior distribution with linear adjustment. The adjustment pushes the posterior closer to the true value of the parameter and overall squash the posterior distribution. However, the true value of the ψ_2 is slightly out of the credible interval. The lowest $RMSE_{\hat{\theta}}$ is achieved by OLS and NPSML, however, the difference with other methods is negligible. It is a good sign that our methodology is able to match the performance of the traditional estimator OLS on this simple time-series. Overall, the methods prove their functionality. It is interesting to note that the $RMSE_{\hat{\theta}}$ of the ABC before linear adjustment is 0.0141 equal to the SMM.

6.2 ARMA(1,1)-GARCH(1,1)

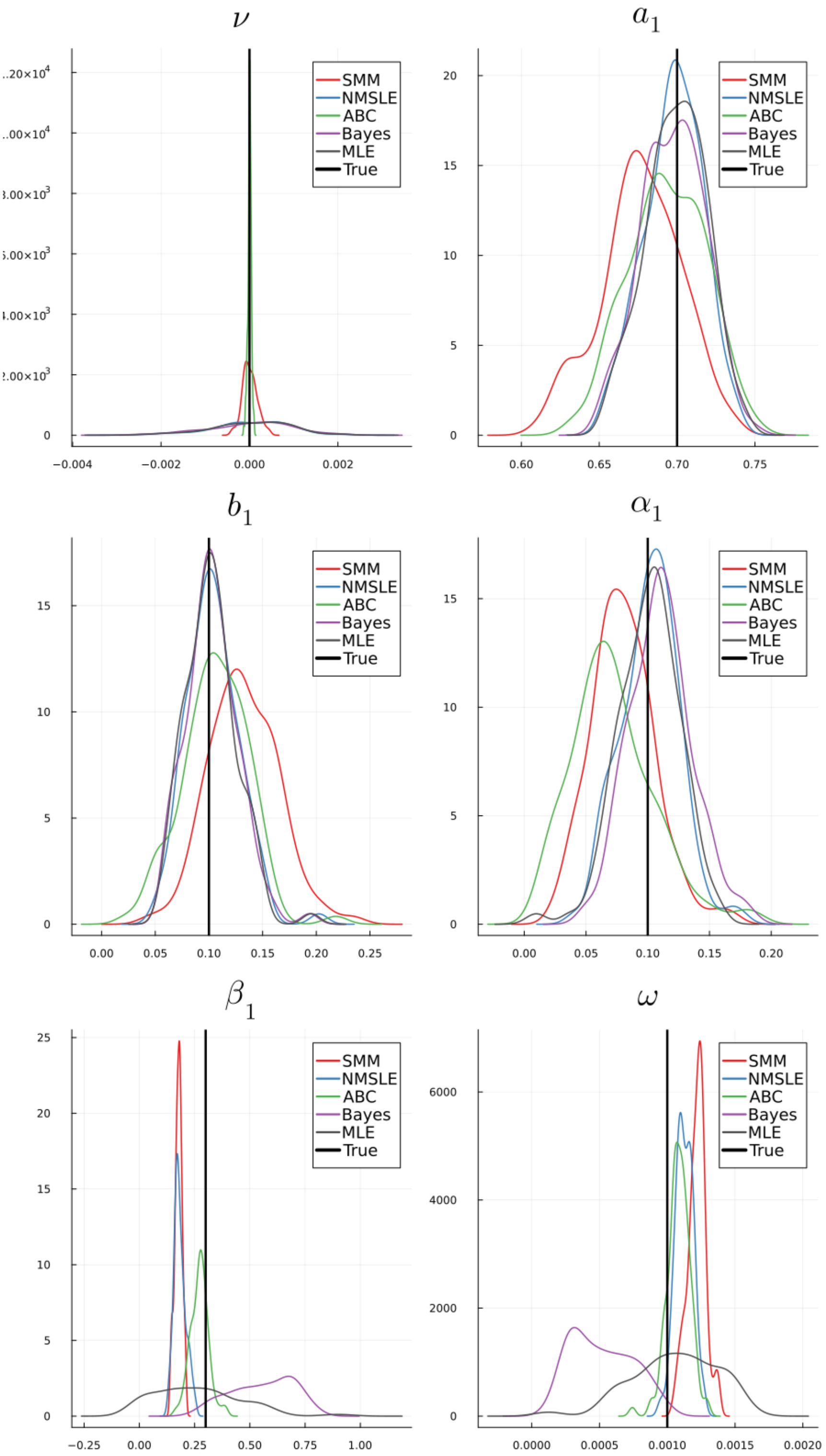
The second baseline model provides us with more complicated dynamics, the ARMA(1,1)-GARCH(1,1) consists of a latent variable and, therefore, is closer to our primary model of interest, Franke & Westerhoff (2012). Another reason why we chose this model is that it is normally estimated with classical econometrics approaches, which we can use as a benchmark for our methods.

The density of the estimated parameters is demonstrated in Figure 6.2, and the averaged results are presented in Table 6.2. For the ARMA paramteres, ν, a_1, and b_1, there is no notable difference between the likelihood methods (BE, NPSML, and MLE). However, for the ν parameter, the densities from SMM and ABC are narrower. The ABC then acts similarly to the likelihood methods in the case of the a_1 and b_1 parameters. Whereas the SMM slightly underestimates the a_1 and overestimates the b_1. Nevertheless, the true values of all three parameters lie inside the 95% confidence/credible interval given by each method.

Moving to the GARCH parameters, the α_1 is estimated almost identically by MLE, NPSML, and BE. The SMM and ABC underestimate a little bit the parameter. However, its true values lie inside the 95% confidence/credible interval. The β_1 parameter is where the methods are having the biggest problem. It is, on average, underestimated by all methods except BE, which overestimates the parameter. The closest estimate is by ABC, followed by MLE, which has a broader density of estimated values. However, the 95% confidence/credible interval of each method contains the true value of β_1, but the intervals are very broad. Finally, the baseline level of volatility ω is reasonably estimated by each method except the BE, which estimates circa half a true value. However, the densities indicate a slight upward bias from the other three simulation-based methods.

It is surprising that in terms of the $RMSE_{\hat{\theta}}$ the MLE is only beating the BE. ABC achieves the lowest value followed with NPSML. It is important to remind

Figure 6.2: ARMA(1,1)-GARCH(1,1): Density of estimated parameters



Note: Kernel density plot of the mean estimates from each of the 100 independent simulations. `ARCHModels.jl` package is used to obtain the MLE estimates.

that the $RMSE_{\hat{\boldsymbol{\theta}}}$ takes into account the bias of the estimate, but the variance as well. The reason why three methods achieve lower $RMSE_{\hat{\boldsymbol{\theta}}}$ than MLE is within parameters β_1 and ω, where the MLE in average achieve better estimate. However, when examining the density plot, the MLE estimates are more spread, whereas our methodology has narrower densities with slight bias. Therefore, even when the MLE has almost no bias in estimation unlike our methodology, the deviation of the estimates is what is increasing its $RMSE_{\hat{\boldsymbol{\theta}}}$. Although, the simulation methods indicates slight bias in comparison to MLE, they are still able to almost match its performance, proving their correctness within the latent variable environment. The fact that BE and NPSML is behaving almost identically to MLE for most of the parameters demonstrate that the approximation of the likelihood function from the simulation methods should be very close to the one from traditional approach.

Table 6.2: ARMA(1,1)-GARCH(1,1): Simulation results using mean estimates

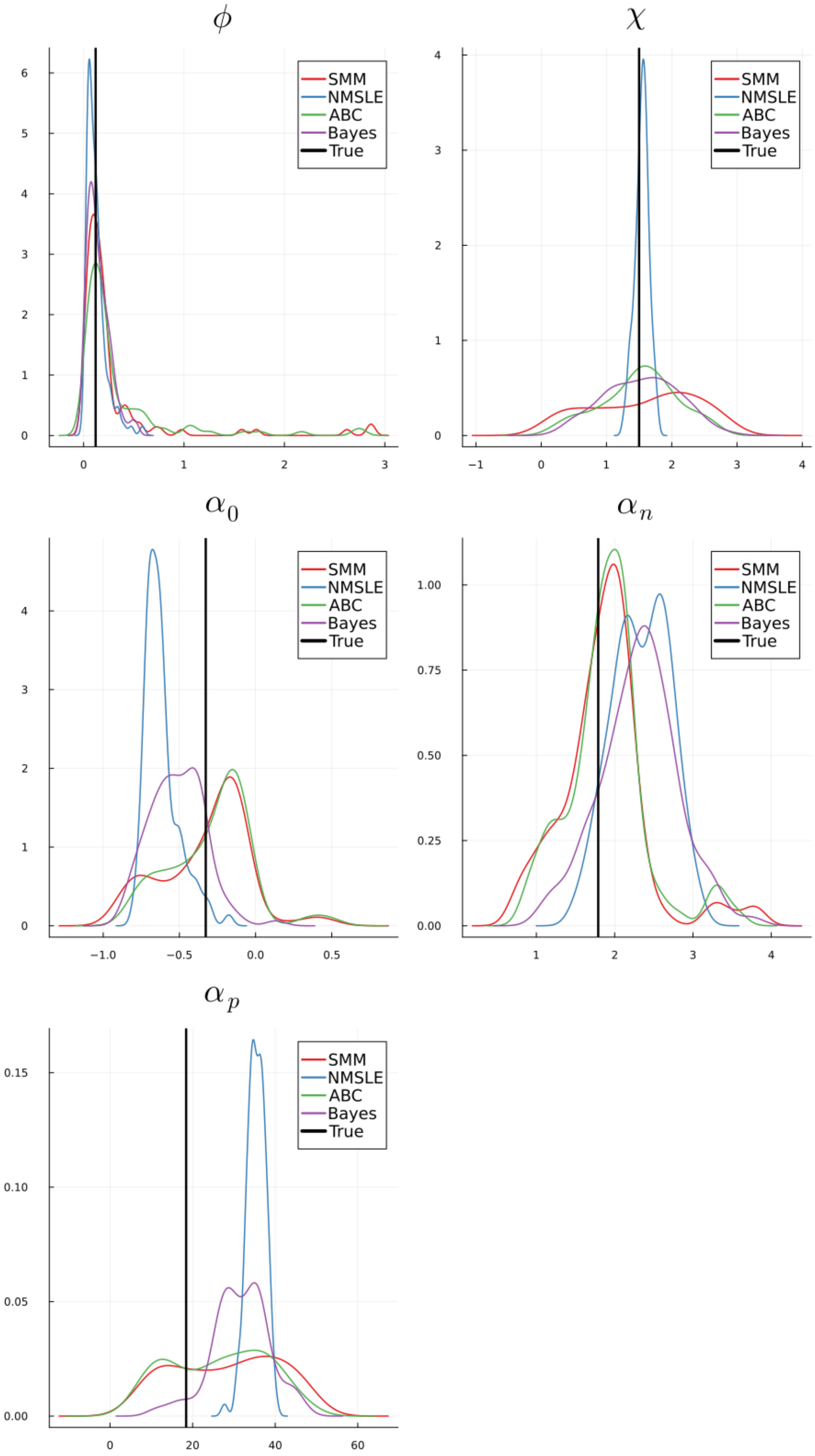
	θ_{true}	SMM	NPSML	ABC	BE	MLE
ν	0	0.0000	0.0000	0.0000	0.0000	0.0001
		(-0.0031, 0.0031)	(-0.0012, 0.0013)	(-0.0022, 0.0022)	(-0.0015, 0.0016)	(-0.0015, 0.0016)
a_1	0.7	0.6770	0.6976	0.6935	0.6975	0.6994
		(0.5991, 0.7352)	(0.6700, 0.7248)	(0.6600, 0.7273)	(0.6643, 0.7304)	(0.6650, 0.7339)
b_1	0.1	0.1323	0.1038	0.1061	0.1031	0.1020
		(0.0445, 0.2629)	(0.0630, 0.1436)	(0.0556, 0.1551)	(0.0552, 0.1501)	(0.0525, 0.1515)
α_1	0.1	0.0803	0.1020	0.0720	0.1105	0.1009
		(0.0074, 0.1742)	(0.0663, 0.1475)	(0.0149, 0.1398)	(0.0706, 0.1536)	(0.0533, 0.1486)
β_1	0.3	0.1768	0.1828	0.2719	0.5575	0.2645
		(0.0058, 0.5350)	(0.0078, 0.5501)	(0.0103, 0.7360)	(0.1908, 0.8092)	(-0.1135, 0.6425)
ω	0.001	0.0012	0.0011	0.0011	0.0005	0.0011
		(0.0006, 0.0016)	(0.0005, 0.0014)	(0.0004, 0.0015)	(0.0002, 0.0011)	(0.0005, 0.0017)
$RMSE_{\hat{\boldsymbol{\theta}}}$	-	0.2537	0.2332	0.2271	0.2901	0.2582

Note: The table presents average mean estimates of the ARMA(1,1)-GARCH(1,1) model's parameters over 100 independent simulations. The average 95% sample confidence/credible intervals of the sample estimates are inside () brackets. `ARCHModels.jl` package is used to obtain the MLE estimates and confidence intervals.

6.3 Franke and Westerhoff (2012)

The third and final model is the Franke & Westerhoff (2012) as the representative of the financial ABM. It is the main point of interest to us. The comparison of the performance on this model is even more critical when we consider that, to our knowledge, there is no study of the frequentist NPSML on the Franke & Westerhoff (2012) model. Therefore, we attempt to yield insight into the

Figure 6.3: Franke and Westerhoff (2012): Density of estimated parameters



Note: Kernel density plot of the mean estimates from each of the 100 independent simulations.

performance in this type of environment while comparing it with SMM, which is more conventional in the financial ABM and is already successfully used to estimate this model (Franke & Westerhoff 2012; Zila & Kukacka 2023), and with their Bayesian counterparts. Note that the results for ABC are without the linear adjustment, as it do not improve the results.

The density of the estimates of parameters from each simulation run is demonstrated in Figure 6.3, and the averaged results over the simulation runs are presented in Table 6.3. At first glance, we can see that the adjustment rate of the fundamentalists' demand ϕ is the parameter with the best results. The densities from each method are mainly concentrated around the true value. However, when we look at Table 6.3 on the averaged values, we can notice that the SMM and ABC overestimate the parameter and that the true value is outside of the 95% confidence/credible intervals which are averaged over the simulation runs. As can be seen in the density plot, SMM and ABC tend to overestimate the parameter in a few simulation runs, increasing the bias in the averaged values. However, the likelihood methods, BE and NPSML, provide a more stable fit of the ϕ without the overestimation that occurs with simulated moments methods. The true value lies inside the 95% confidence/credible interval, with the credible interval from BE being slightly narrower.

Estimating the adjustment rate of the chartists' demand χ can be seen as a success if we only inspect the average estimate. However, all the 95% confidence/credible intervals are very broad. This indicates that within the single simulation run, the posterior distribution or the point estimates are spread around the whole range of possible values. Indeed, during all of our estimation exercises, none of the methods can capture the χ parameter with some consistency.

The predisposition α_0 is better captured by the SMM and ABC. However, the density plot is not evenly distributed around the true value. The likelihood methods seem to be underestimating the parameter. What is positive is that the true value of the parameter is inside the 95% confidence/credible interval for all methods. Furthermore, the SMM and ABC can also provide better estimates for the herding parameter α_n, with density plots concentrated around the parameter's true value. The NPSML and BE tend to overestimate the α_n. Nevertheless, the true value of the parameter lies inside the 95% confidence/credible interval for each method except for BE.

The price misalignment α_p is, on average, overestimated by each method. However, the likelihood methods and the simulated moments methods differ.

Table 6.3: Franke and Westerhoff (2012): Simulation results using mean estimates

	θ_{true}	SMM	NPSML	ABC	BE
ϕ	0.12	0.2843	0.1231	0.3433	0.1468
		(0.1638, 0.5104)	(0.0272, 0.2753)	(0.1380, 0.6992)	(0.0638, 0.2469)
χ	1.5	1.5679	1.5431	1.5082	1.5362
		(0.5111, 2.5369)	(0.0987, 2.9080)	(0.2534, 2.7120)	(0.6277, 2.4416)
α_0	-0.327	-0.3240	-0.6180	-0.2883	-0.5137
		(-0.5506, -0.1488)	(-0.9671, -0.2532)	(-0.5331, -0.0891)	(-0.7426, -0.2987)
α_n	1.79	1.8635	2.3360	1.9045	2.3097
		(1.5156, 2.2290)	(1.5011, 3.2812)	(1.5122, 2.2616)	(1.8698, 2.8041)
α_p	18.43	28.1715	35.3574	26.2287	31.6034
		(17.3187, 39.4321)	(12.6718, 49.3757)	(13.7120, 38.9413)	(17.9033, 42.6307)
$RMSE_{\hat{\boldsymbol{\theta}}}$	-	1.3715	1.0776	1.2401	1.0501

Note: The table presents average mean estimates of the Franke & Westerhoff (2012) model's parameters over 100 independent simulations. The average 95% sample confidence/credible intervals of the sample estimates are inside () brackets.

Where the ABC and SMM estimates are spread around the whole possible interval, the densities of the NPSML and BE are narrower but above the true value of the parameter. The averaged 95% confidence/credible intervals are very broad within each method. Consequently, they all contain the true value of the α_p.

Likelihood methods achieve lower $RMSE_{\hat{\boldsymbol{\theta}}}$ with the BE having the edge over the NPSML. The ABC has lower $RMSE_{\hat{\boldsymbol{\theta}}}$ than the SMM. The Bayesian methods prove their rise in the literature is justified as they outperform their frequentist counterparts. Overall, the difference in performance between methods is not that big. However, the performance between methods differs per parameter. The moment's methods slightly outperform the likelihood methods in terms of bias in 3 out of 5 parameters, and one of the remaining parameters is not estimated by any of the methods. On the other hand, the estimates from the moment methods are more widespread, and the deviation of the estimates is the primary reason why they have higher $RMSE_{\hat{\boldsymbol{\theta}}}$. It seems like the decision between the likelihood or moment method is similar to the bias-variance tradeoff. In the case of the Franke & Westerhoff (2012) model, if the researcher seeks lower bias and does not mind the higher variance, he should choose the moment method according to our study. The researcher is always free to conduct more replications of the estimation for the cost of another computational resource to deal with the higher variance. On the other hand, one of the important aspects of our study is the establishment of the biases for given methods. Using this knowledge, the researcher can then, in the empirical study, treat the estimates as the upper or lower bound of the parameter based on the direction of the given bias. Therefore, the likelihood methods seem to provide an appealing alternative to the moment's methods, as the behavior of the likelihood methods seems to be more stable over different random seeds. An interesting exercise can be model averaging, utilizing the knowledge of the method's performance and using it for, e.g., a weighted average of the estimates, which could be superior to the independent methods. However, it is out of the scope of this thesis.

Chapter 7

Discussion

The chapter discusses additional exercises to determine the limitations and further improvements in the implemented methods. Section 7.1 starts with a sensitivity analysis to the magnitude of the noise. Subsequently, Section 7.2 moves to robustness to results when the parameters are derived as median or mode instead of mean. Thirdly, Section 7.3 depicts the trajectory within the likelihood calculation. Lastly, Section 7.4 concludes the chapter with the caveats of the linear adjustment.

7.1 Sensitivity Analysis

The parametrization of the Franke & Westerhoff (2012) model follows the original study. However, we examine the sensitivity of the estimation methods to the different magnitudes of the standard deviation of the noise terms by scaling the standard deviations by factors 5 and 1/5. More precisely, the standard deviation for the small noise variant is set such that $\sigma_f^{small} = 0.1516$ and $\sigma_c^{small} = 0.4174$. In the case of the large noise alternative, the deviations are $\sigma_f^{large} = 3.79$ and $\sigma_c^{large} = 10.435$. All other parameters have the same value, and the optimization constraints and uniform biases are unchanged.

The large and small noise results are presented in Table 7.1 and Table 7.2, respectively. The SMM and ABC have worsened in terms of $RMSE_{\hat{\boldsymbol{\theta}}}$ in both variants, indicating that the moments matching becomes difficult with different noise settings. The NPSML and BE achieve slightly better $RMSE_{\hat{\boldsymbol{\theta}}}$ for large noise variant with NPSML having the lowest value. The $RMSE_{\hat{\boldsymbol{\theta}}}$ is higher for the small noise alternative. However, it is lower for likelihood methods than for moment ones.

The adjustment rate of the fundamentalists' demand ϕ is overestimated substantially by SMM and ABC with comparison to results in the Table 6.3. Whereas the NPSML and BE also show slight overestimation, the true value of the parameter still lies inside the 95% confidence/credible interval. Estimating the adjustment rate of the chartists' demand χ shows the same behavior by all of the methods as in Table 6.3.

With the predisposition α_0, there is not much difference in performance, only that it is slightly overestimated by SMM and ABC in the large noise variant and underestimated in the small noise variant. The NPSML and BE estimates are similar to the standard case. The true value of the α_0 lies inside each of the 95% confidence/credible intervals. The same goes for the herding α_n, where the estimates for large noise are similar to the standard case, and for the small noise, the estimates are slightly smaller. In case of the price misalignment, α_p, it is still heavily overestimated. However, each of the 95% confidence/credible intervals contains the true value of the parameter. Overall, the likelihood methods show higher robustness to the different noise magnitudes as their results are similar to the standard version. It is important to note that the likelihood methods assume that the noise's distribution is known, which is why the parameters of the noise's volatility are fixed; with moments methods, we can relax it.

Table 7.1: Franke and Westerhoff (2012): Simulation results using mean estimates, large noise

	θ_{true}	SMM	NPSML	ABC	BE
ϕ	0.12	0.7926	0.1690	0.9109	0.2005
		(0.4183, 1.2685)	(0.0233, 0.3840)	(0.3553, 1.7035)	(0.0898, 0.3285)
χ	1.5	1.5347	1.5053	1.5175	1.5439
		(0.2316, 2.8130)	(0.0861, 2.9070)	(0.1265, 2.8862)	(0.6462, 2.4746)
α_0	-0.327	-0.2316	-0.5438	-0.2151	-0.4981
		(-0.4937, -0.0269)	(-0.9141, -0.1020)	(-0.5298, 0.0308)	(-0.7738, -0.2533)
α_n	1.79	1.9595	2.0377	1.9914	2.1247
		(1.4316, 2.4282)	(0.8092, 3.3009)	(1.3243, 2.5893)	(1.4614, 2.8986)
α_p	18.43	28.6200	28.4730	27.9975	28.4487
		(18.4215, 38.5448)	(7.6980, 47.8247)	(15.6815, 40.1742)	(15.3749, 41.6454)
$RMSE_{\hat{\boldsymbol{\theta}}}$	-	1.7708	0.8105	1.6523	1.0342

Note: The table presents average mean estimates of the Franke & Westerhoff (2012) model's parameters over 100 independent simulations. The average 95% sample confidence/credible intervals of the sample estimates are inside () brackets. Standard deviations are set such that $\sigma_f^{large} = 3.79$ and $\sigma_c^{large} = 10.435$.

Table 7.2: Franke and Westerhoff (2012): Simulation results using mean estimates, small noise

	θ_{true}	SMM	NPSML	ABC	BE
ϕ	0.12	0.7451	0.2238	0.8038	0.2920
		(0.3488, 1.1669)	(0.0316, 0.5281)	(0.2895, 1.4771)	(0.0492, 0.6655)
χ	1.5	1.5273	1.4886	1.4994	1.5481
		(0.7726, 2.2514)	(0.1594, 2.8485)	(0.4880, 2.4955)	(0.2710, 2.7603)
α_0	-0.327	-0.4549	-0.5335	-0.4145	-0.4604
		(-0.6680, -0.2295)	(-0.8306, -0.2110)	(-0.6783, -0.1541)	(-0.7618, -0.1662)
α_n	1.79	1.5082	1.5101	1.5750	1.6048
		(1.1651, 1.8702)	(0.9520, 2.0102)	(1.1486, 1.9917)	(1.0464, 2.0696)
α_p	18.43	32.8266	40.9245	30.3750	36.4523
		(17.7747, 46.6183)	(23.8870, 49.6887)	(12.8954, 47.3929)	(18.3906, 48.5793)
$RMSE_{\hat{\boldsymbol{\theta}}}$	-	1.7823	1.4614	1.5790	1.2843

Note: The table presents average mean estimates of the Franke & Westerhoff (2012) model's parameters over 100 independent simulations. The average 95% sample confidence/credible intervals of the sample estimates are inside () brackets. Standard deviations are set such that $\sigma_f^{small} = 0.1516$ and $\sigma_c^{small} = 0.4174$.

7.2 Median and Mode Estimates

The choice of the final estimate being the mean is common in the ABM literature (Lux 2022; 2023a; Platt 2022; Zhang *et al.* 2023). This section examines the difference when we consider the median and also, for the Bayesian methods, the mode instead of the mean.

When we consider the median of the posterior distribution or the point estimates as the final estimate within the single estimation run, we present in the Table 7.3. The Table 7.4 demonstrate a similar thing for the mode of the posterior distribution for the ABC and BE. We do not see a significant difference in averaged estimates when we compare the two tables with Table 6.3.

Nevertheless, the $RMSE_{\hat{\boldsymbol{\theta}}}$ are higher for the median and mode estimates. It can be tempting to say that it implies that within the single simulation run, the median and mode can be more biased than the mean, resulting in higher $RMSE_{\hat{\boldsymbol{\theta}}}$. However, it can be moreover dependent on the choice of the metric. $RMSE_{\hat{\boldsymbol{\theta}}}$ is mainly loss driven by squared differences, which are minimized by the mean, whereas the median minimizes the absolute differences and mode the $0 - 1$ loss (Murphy 2012). The mean and median should be preferred in most cases as the mode can lead to atypical points (Murphy 2012; Platt 2020).

Table 7.3: Franke and Westerhoff (2012): Simulation results using median estimates

	θ_{true}	SMM	NPSML	ABC	BE
ϕ	0.12	0.2653	0.1120	0.3156	0.1428
		(0.1638, 0.5104)	(0.0272, 0.2753)	(0.1380, 0.6992)	(0.0638, 0.2469)
χ	1.5	1.5934	1.5551	1.5228	1.5538
		(0.5111, 2.5369)	(0.0987, 2.9080)	(0.2534, 2.712)	(0.6277, 2.4416)
α_0	-0.327	-0.3166	-0.6158	-0.2776	-0.5112
		(-0.5506, -0.1488)	(-0.9671, -0.2532)	(-0.5331, -0.0891)	(-0.7426, -0.2987)
α_n	1.79	1.8687	2.3124	1.9184	2.3032
		(1.5156, 2.2290)	(1.5011, 3.2812)	(1.5122, 2.2616)	(1.8698, 2.8041)
α_p	18.43	28.1727	37.1856	26.2675	31.6417
		(17.3187, 39.4321)	(12.6718, 49.3757)	(13.7120, 38.9413)	(17.9033, 42.6307)
$RMSE_{\hat{\boldsymbol{\theta}}}$	-	1.4334	1.1741	1.3020	1.1398

Note: The table presents average median estimates of the Franke & Westerhoff (2012) model's parameters over 100 independent simulations. The average 95% sample confidence/credible intervals of the sample estimates are inside () brackets.

Table 7.4: Franke and Westerhoff (2012): Simulation results using mode estimates

	θ_{true}	ABC	BE
ϕ	0.12	0.2758	0.1454
		(0.1380, 0.6992)	(0.0638, 0.2469)
χ	1.5	1.5646	1.5542
		(0.2534, 2.7124)	(0.6277, 2.4416)
α_0	-0.327	-0.2531	-0.5185
		(-0.5331, -0.0891)	(-0.7426, -0.2987)
α_n	1.79	1.9383	2.2870
		(1.5122, 2.2616)	(1.8698, 2.8041)
α_p	18.43	26.3883	32.4774
		(13.7120, 38.9413)	(17.9033, 42.6307)
$RMSE_{\hat{\boldsymbol{\theta}}}$	-	1.5041	1.2278

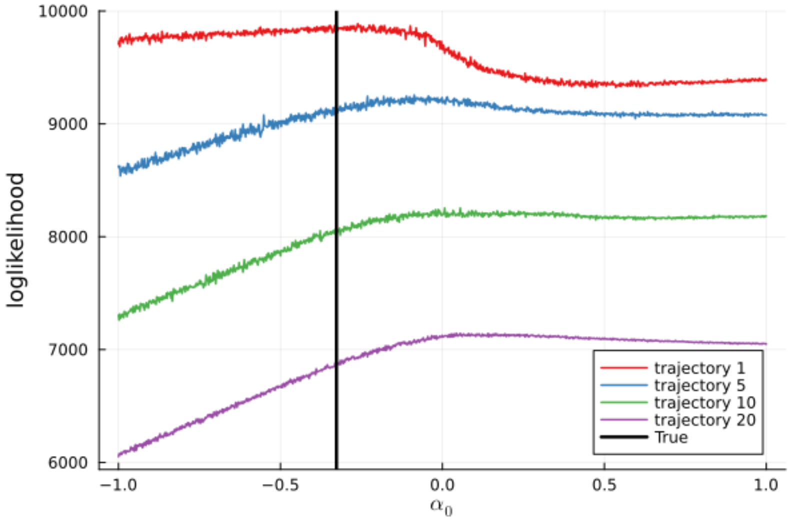
Note: The table presents average mode estimates of the Franke & Westerhoff (2012) model's parameters over 100 independent simulations. The average 95% sample confidence/credible intervals of the sample estimates are inside () brackets.

7.3 Trajectory for the Likelihood Approximation

We intend to shed more light on the trajectory utilized in the likelihood calculation for models with latent variables, depicted in Subsection 3.2.3. As stated in Section 5.2 during our initial simulation exercises, it shows that it is not behaving as expected. The best-performing trajectory is of size 1. However, it practically turns the Subsection 3.2.3 into Subsection 3.2.1 with an additional burn-in period of the given model to derive the vector of latent variables. This approach would need further research if the theoretical properties are still satisfied, but it is out of the scope of this thesis.

Our implementation follows Kristensen & Shin (2012) and Creel & Kristensen (2012). The trajectory implementation is based on the extension to a non-stationary series with a latent variable in Creel & Kristensen (2012). The robustness to non-stationarity is why we can stick to the original log price with the Franke & Westerhoff (2012) model even when the time series is not stationary. We also try the approach with log returns, which makes the time series stationary; however, for the trajectory of size 1, the approach is equivalent to the one with log prices, and for higher trajectories, it has the same or worse performance.

Figure 7.1: Loglikelihoods of the parameter α_0 in the Franke and Westerhoff (2012) model with different trajectory sizes.

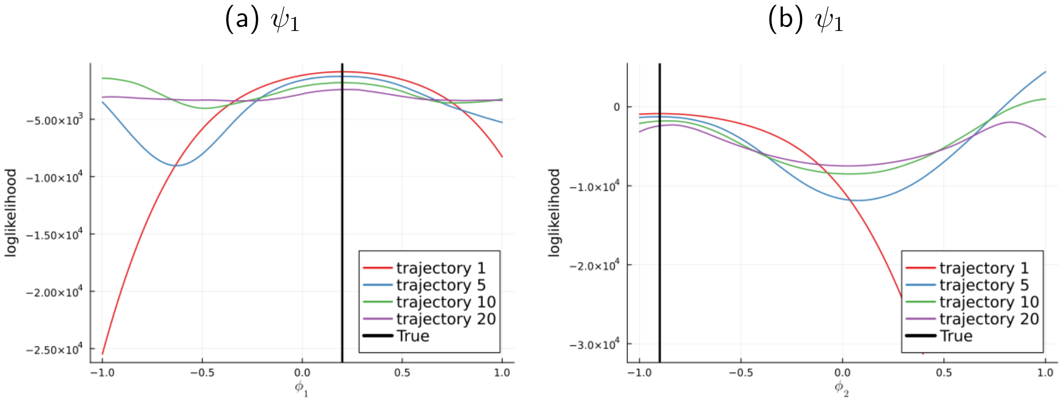


Note: For each trajectory, the loglikelihood is calculated for 1000 values of α_0, equally spaced in the [-1,1] interval, while keeping other parameters fixed.

The Figure 7.1 shows the loglikelihood for predisposition parameter α_0 inside the Franke & Westerhoff (2012) model while keeping other parameters fixed at their true value. For the trajectory of size 1, we can see that the loglikelihood is almost flat for negative values with a maximum close to the true value, and the loglikelihood drops for positive values of α_0. When the trajectory size increases, the function's behavior changes in that it is not so flat within negative values; however, there is no drop within positive values. Nevertheless, the maximum of the likelihood is moving to the right side further from the true value with the increasing trajectory size, which introduces bias into the estimation procedure.

We assume that the trajectory extension for NPSML should work even with models without latent variables. Hence, we test it on the simplest model we use in our study, AR(2). Independently calculating the loglikelihood for both parameters while keeping the second one fixed at true value, the result is in the Figure 7.2. For trajectory of size 1, the loglikelihood is well behaved for both parameters; however, with the increasing trajectory size, the loglikelihood function starts to change into the bimodal function, and for higher values, flatting across the range of the possible values. Therefore, we believe the KDE and the trajectory setting when dealing with latent variables need further research.

Figure 7.2: Loglikelihoods of the parameters in the AR(2) model with different trajectory sizes.


(a) ψ_1

(b) ψ_1

Note: The loglikelihoods are calculated separately for each parameter in a way that for each trajectory, the loglikelihood is calculated for 200 values of the given parameter, equally spaced in the [-1,1] interval, while keeping other parameters fixed.

From our understanding, the main problem does not lie within the trajectory itself but the conditioning set as it disturbs the KDE. It can be interesting to use the trajectory without the conditioning set, meaning that for KDE approximation of the likelihood is used only the last value from the trajectory for a given time stamp. Another approach can be from the latent variable extension in Kristensen & Shin (2012). Note, that our implementation has always start of the trajectory inside the empirical time series, and separate trajectories are simulated for each time stamp. Whereas the Kristensen & Shin (2012) recommend simulating the whole time series, the so-called long trajectory. It reminds the approach inside SMM, where the whole time series are simulated similarly; nevertheless, the simulated time series are not compared to the empirical time series using moments but by the KDE methods to derive the likelihood. Note that this approach needs a stationary series, as the simulated data are not grounded in the empirical series. Therefore, it needs the data to fluctuate around a stationary level (we utilize log-returns within Franke & Westerhoff (2012) model). The researcher is determining the size of the conditioning set for KDE, but the approximation of the likelihood with the conditioning set is behaving even worse than the one described above. However, the long trajectory without conditioning set can compete with our approach and can obtain meaningful results, though, with worse $RMSE_{\hat{\boldsymbol{\theta}}}$ than our approach. Note that the long trajectory without conditioning set is technically the approach used for likelihood approximation within Bayesian estimation in Grazzini *et al.* (2017). Platt (2022) utilizes the Bayesian estimation with this approach on Brock & Hommes (1997) and Franke & Westerhoff (2012) models; however, the study focuses more on likelihood approximation with neural networks.

The limitation in our final approach can be seen that as the initial values of the latent variable, we derive the vector of these values using multiple burn-in periods from the model. When the burn-in period is sufficiently long, it should give us good representative values. As the likelihood is calculated from N_L simulation, we obtain a vector of N_L values for the latent variable. Therefore, it can be seen as a distribution of possible values of the latent variable with a given parameter set. Note that each time stamp uses this vector of initial values for latent variables during likelihood calculation. The whole point of the trajectory simulation is then to give time to the latent variable to settle with the current dynamics of the model. However, this is redundant when we find out that the best-performing setting is the trajectory of size 1 (it can also be seen as no trajectory). There is a possible inspiration inside the classical econometrics when dealing with latent variables; when estimating the GARCH parameters, the traditional approach is to determine the initial values of the σ and then update it step by step as the likelihood is calculated (Francq & Zakoian 2004). These online updates of the latent variable can be an interesting extension of our current approach to the likelihood calculation; unfortunately, we leave it for further research due to computational and time constraints.

7.4 Linear Adjustment

The linear adjustment proves to substantially improve the results of the ABC for the AR(2) model. Unfortunately, we have to omit the adjustment in case of more complex models, ARMA(1,1)-GARCH(1,1) and Franke & Westerhoff (2012). When the posterior distribution is far from a normal distribution, e.g., multimodal, weirdly skewed to the one side, then in our experiments, the linear adjustment do not work as intended, pushing the posterior further away from the true value. The intercept from the regression mainly determines the adjustment. It acts as the main point around which the adjusted posterior is shaped. However, when the initial posterior is not already equally spread around some value, then the intercept from the regression tends to be estimated outside of the current posterior, often far away from the true value of the parameter, pushing the adjusted posterior even behind the bounds of the initial uniform prior. It is important to note that circa in half of the simulation runs with the Franke & Westerhoff (2012) model, the linear adjustment can improve the final posterior. However, the overall instability of the approach forced us to omit it from the final results.

We try multiple approaches to deal with this problem during the development phase. Lux (2023a) also presents a non-linear extension to control heteroskedasticity, but the intercept is estimated similarly, thus facing the same problems as the simple linear adjustment. To mitigate the problem of the strictly positive parameters being negative after adjustment, we utilize the log transformation of particles depicted in Blum (2017), then use the linear adjustment on the log particles and exponentiate them back. In our exercises, it results in extreme values. Subsequently, we try different transformations to reflect the uniform priors; we first move the particles into the $[0, 1]$ range based on the bounds of the uniform prior, then transform the particles using the inverse of the logistic function, adjust the transformed particles, and transforming back the adjusted version of the particles. However, this approach results in the adjusted posteriors being concentrated to the bound of the uniform prior. Lastly, we follow Blum (2017) with non-linear correction using a small neural network. Nevertheless, different initializations of the neural networks can provide different results; the corrected posterior can be superior to other approaches or, worst of all, making the whole procedure very unstable.

Therefore, we decide to stick to the simplest linear adjustment and present it only in the case of the AR(2) model to demonstrate its potential in improving the results from ABC, and further research is needed in the case of the financial ABM.

Chapter 8

Conclusion

We conduct a comparison study of simulation-based estimation methods focusing on the financial ABMs. These models are gaining in popularity as they can replicate the stylized facts of the financial markets. However, their analytical solution is often infeasible, forcing the researchers to abandon the traditional approaches, such as least squares or maximum likelihood. The most commonly used approaches are simulation-based methods. More precisely, we selected four prominent methods consisting of SMM, NPSML, ABC, and BE. The SMM is widely the most used estimation technique in financial ABMs; its implementation is straightforward with known theoretical properties. However, it hugely depends on the selection of moments.

On the other hand, more novel NPSML is by simulating multiple values from the model approximating the conditional density with KDE methods and using this information for likelihood calculation. Although the selection of moments is missing, the NPSML comes with the assumption of the known distribution of the noise term. We add two methods from the Bayesian framework to these two frequentist methods. ABC is popular in the biological modeling field, and it shows promising results in the financial ABMs. The method can be seen as the Bayesian counterpart of the SMM because it compares the simulated moments to the empirical ones to assess the quality of a given particle. We use the SMC algorithm to derive the posterior. Lastly, we implement BE with likelihood approximation being the same as in NPSML, and again, the posterior is obtained using SMC algorithm. Therefore, we present a set of likelihood and likelihood-free estimation techniques based on frequentist and Bayesian versions.

Our contribution to the literature is not only in the set of the selected simulation-based estimation techniques. Our performance comparison mainly focuses on the financial ABM with latent variable. The latent variable is no problem for the simulation moment techniques; however, in the case of the simulated likelihood, it is not so straightforward. We believe we are among the first to assess the performance of the frequentists NPSML on the financial ABM with latent variable. As the benchmarking models, we select two traditional econometric models, AR(2) model and ARMA(1,1)-GARCH(1,1) model, the leading benchmark being the popular Franke & Westerhoff (2012) model.

The AR(2) model serves only as the simple baseline every method should pass. All four stated methods can correctly estimate the parameters of AR(2), and their performance is identical to the traditional OLS. The ARMA(1,1)-GARCH(1,1) model provides us with more complex dynamics. The model consists of a latent variable. However, it is typically estimated using traditional techniques. It demonstrates that all four methods can deal with latent variables and can be close to the performance of the classical MLE. It is interesting that the parameters of the ARMA part and the parameter controlling the effect of the past noise on current volatility we estimate almost identically by MLE, NPSML, and BE with SMM and ABC being not so different. However, the estimation of the baseline level of volatility and the parameter controlling the effect of past volatility shows a slight bias.

The main focus is on the estimation of the Franke & Westerhoff (2012) as the estimation of this model is not feasible with analytical methods. Therefore, it is the primary model where the stated methods are used. When assessing the performance purely just by the $RMSE_{\hat{\boldsymbol{\theta}}}$ the likelihood methods are outperforming the moment methods with the Bayesian variants beating the frequentists counterparts. Similarly to the Platt (2020), BE arrives on top. However, it is essential to assess the method parameter by parameter. The estimation of the adjustment rate of the chartists seems to act as a uniform distribution over the given constraint. Meanwhile, the estimation of the adjustment rate of the fundamentalist is more successful, with the likelihood methods being consistently better. On the other hand, the predisposition and herding parameters are better estimated with moments methods, while the likelihood methods show a bias in the estimation. Lastly, the mispricing parameter is overestimated by all methods, more by the likelihood methods. Therefore, by the parameter per parameter assessment, it seems that the ABC and SMM provide less biased results for the majority of the parameters than NPSML and BE; however, the likelihood methods has lower variance of their estimates which is the primary reason why they achieve lower $RMSE_{\hat{\boldsymbol{\theta}}}$.

The results on the Franke & Westerhoff (2012) suggest that the increase in the usage of the Bayesian techniques in the financial ABMs is reasonable as they propose a competitive alternative to the frequentist methods. Although BE achieves the lowest results in terms of $RMSE_{\hat{\boldsymbol{\theta}}}$, the likelihood approximation used by NPSML also needs further investigation in the environment with latent variable to tackle down the bias in the estimation. Precisely, the trajectory for the likelihood approximation needs to be more closely studied. It is also possible to take inspiration from the MLE version used for the GARCH models. The second Bayesian technique, ABC, provides a compelling alternative to the SMM; however, we are not able to utilize the regression adjustment of the final posterior, and we believe that further research in the possible adjustments can improve the performance of the ABC.

The choice between the likelihood and moments methods seems to be based on a bias-variance tradeoff, where the moments methods demonstrate lower bias on average; however, there is a higher variance in the estimates than the likelihood methods. Researchers can utilize the knowledge from our simulation study to establish the approximate biases in given parameters and utilize it in empirical study to obtain the lower or upper bound on the given parameter from the estimated parameters. Further research can be interesting in model averaging, as the weighted average from the given methods could be superior to the independent methods.

Bibliography

Alfarano, S., T. Lux, & F. Wagner (2005): “Estimation of agent-based models: the case of an asymmetric herding model.” *Computational Economics* **26**: pp. 19–49.

Alfarano, S., T. Lux, & F. Wagner (2008): “Time variation of higher moments in a financial market with heterogeneous agents: An analytical approach.” *Journal of Economic Dynamics and Control* **32(1)**: pp. 101–136.

Allen, H. & M. P. Taylor (1990): “Charts, noise and fundamentals in the london foreign exchange market.” *The Economic Journal* **100(400)**: pp. 49–59.

Altissimo, F. & A. Mele (2009): “Simulated non-parametric estimation of dynamic models.” *The Review of Economic Studies* **76(2)**: pp. 413–450.

Altonji, J. G. & L. M. Segal (1996): “Small-sample bias in gmm estimation of covariance structures.” *Journal of Business & Economic Statistics* **14(3)**: pp. 353–366.

Bandi, F. M. & P. C. Phillips (2003): “Fully nonparametric estimation of scalar diffusion models.” *Econometrica* **71(1)**: pp. 241–283.

Beaumont, M. A., J.-M. Cornuet, J.-M. Marin, & C. P. Robert (2009): “Adaptive approximate bayesian computation.” *Biometrika* **96(4)**: pp. 983–990.

Bertschinger, N. & I. Mozzhorin (2021): “Bayesian estimation and likelihood-based comparison of agent-based volatility models.” *Journal of Economic Interaction and Coordination* **16(1)**: pp. 173–210.

Blum, M. G. (2017): “Regression approaches for approximate bayesian computation.”

Bollerslev, T. (1986): "Generalized autoregressive conditional heteroskedasticity." *Journal of econometrics* **31(3)**: pp. 307–327.

Bornholdt, S. (2001): "Expectation bubbles in a spin model of markets: Intermittency from frustration across scales."

Boswijk, H. P., C. H. Hommes, & S. Manzan (2007): "Behavioral heterogeneity in stock prices." *Journal of Economic dynamics and control* **31(6)**: pp. 1938–1970.

Brock, W. A. & C. H. Hommes (1997): "A rational route to randomness." *Econometrica: Journal of the Econometric Society* pp. 1059–1095.

Campbell, J. Y., A. W. Lo, A. C. MacKinlay, & R. F. Whitelaw (1998): "The econometrics of financial markets." *Macroeconomic Dynamics* **2(4)**: pp. 559–562.

Chen, Z. & T. Lux (2018): "Estimation of sentiment effects in financial markets: A simulated method of moments approach." *Computational Economics* **52(3)**: pp. 711–744.

Chopin, N. (2004): "Central limit theorem for sequential Monte Carlo methods and its application to Bayesian inference." *The Annals of Statistics* **32(6)**: pp. 2385 – 2411.

Cobb, L. (1981): "Parameter estimation for the cusp catastrophe model." *Behavioral Science* **26(1)**: pp. 75–78.

Cont, R. (2001): "Empirical properties of asset returns: stylized facts and statistical issues." *Quantitative finance* **1(2)**: p. 223.

Cont, R. (2007): "Volatility clustering in financial markets: empirical facts and agent-based models." In "Long memory in economics," pp. 289–309. Springer.

Creel, M. & D. Kristensen (2012): "Estimation of dynamic latent variable models using simulated non-parametric moments." *The Econometrics Journal* **15(3)**: pp. 490–515.

De Jong, E., W. F. Verschoor, & R. C. Zwinkels (2009): "A heterogeneous route to the european monetary system crisis." *Applied Economics Letters* **16(9)**: pp. 929–932.

DEL MORAL, P., A. DOUCET, & A. JASRA (2006): "Sequential monte carlo samplers." *Journal of the Royal Statistical Society Series B: Statistical Methodology* **68(3)**: pp. 411–436.

DING, Z., C. W. GRANGER, & R. F. ENGLE (1993): "A long memory property of stock market returns and a new model." *Journal of empirical finance* **1(1)**: pp. 83–106.

DROVANDI, C. C. & A. N. PETTITT (2011): "Estimation of parameters for macroparasite population evolution using approximate bayesian computation." *Biometrics* **67(1)**: pp. 225–233.

DUFFIE, D. & K. J. SINGLETON (1990): "Simulated moments estimation of markov models of asset prices."

EVANS, G. & S. HONKAPOHJA (2001): "Expectations, economics of." In N. J. SMELSER & P. B. BALTES (editors), "International Encyclopedia of the Social & Behavioral Sciences," pp. 5060–5067. Oxford: Pergamon.

FAMA, E. F. (1965): "The behavior of stock-market prices." *The journal of Business* **38(1)**: pp. 34–105.

FEARNHEAD, P. (2018): "Asymptotics of abc." In "Handbook of Approximate Bayesian Computation," pp. 269–288. Chapman and Hall/CRC.

FRANCQ, C. & J.-M. ZAKOIAN (2004): "Maximum likelihood estimation of pure garch and arma-garch processes." *Bernoulli* **10(4)**: pp. 605–637.

FRANKE, R. (2008): "A microfounded herding model and its estimation on german survey expectations." *European Journal of Economics and Economic Policies* **5(2)**: pp. 301–328.

FRANKE, R. (2009): "Applying the method of simulated moments to estimate a small agent-based asset pricing model." *Journal of Empirical Finance* **16(5)**: pp. 804–815.

FRANKE, R. & F. WESTERHOFF (2011): "Estimation of a structural stochastic volatility model of asset pricing." *Computational Economics* **38**: pp. 53–83.

FRANKE, R. & F. WESTERHOFF (2012): "Structural stochastic volatility in asset pricing dynamics: Estimation and model contest." *Journal of Economic Dynamics and Control* **36(8)**: pp. 1193–1211.

FRANKE, R. & F. WESTERHOFF (2016): "Why a simple herding model may generate the stylized facts of daily returns: explanation and estimation." *Journal of Economic Interaction and Coordination* **11**: pp. 1–34.

FRANKEL, J. A. & K. A. FROOT (1990): "Chartists, fundamentalists, and trading in the foreign exchange market." *The American Economic Review* **80(2)**: pp. 181–185.

FRAZIER, D. T., G. M. MARTIN, C. P. ROBERT, & J. ROUSSEAU (2018): "Asymptotic properties of approximate bayesian computation."

GATTI, D. D. & J. GRAZZINI (2020): "Rising to the challenge: Bayesian estimation and forecasting techniques for macroeconomic agent based models." *Journal of Economic Behavior & Organization* **178**: pp. 875–902.

GAUNERSDORFER, A. & C. HOMMES (2007): "A nonlinear structural model for volatility clustering."

GEWEKE, J. (1989): "Bayesian inference in econometric models using monte carlo integration." *Econometrica: Journal of the Econometric Society* pp. 1317–1339.

GILLI, M. & P. WINKER (2003): "A global optimization heuristic for estimating agent based models." *Computational Statistics & Data Analysis* **42(3)**: pp. 299–312.

GRAZZINI, J., M. G. RICHIARDI, & M. TSIONAS (2017): "Bayesian estimation of agent-based models." *Journal of Economic Dynamics and Control* **77**: pp. 26–47.

HERBST, E. & F. SCHORFHEIDE (2014): "Sequential monte carlo sampling for dsge models." *Journal of Applied Econometrics* **29(7)**: pp. 1073–1098.

HIRSHLEIFER, J. (1985): "The expanding domain of economics." *The American Economic Review* **75(6)**: pp. 53–68.

HONG, H. & J. C. STEIN (1999): "A unified theory of underreaction, momentum trading, and overreaction in asset markets." *The Journal of finance* **54(6)**: pp. 2143–2184.

HULL, J. (2012): *Risk management and financial institutions,+ Web Site*, volume 733. John Wiley & Sons.

KARLSEN, H. A. & D. TJØSTHEIM (2001): "Nonparametric estimation in null recurrent time series." *Annals of Statistics* pp. 372–416.

KIRMAN, A. (1993): "Ants, rationality, and recruitment." *The Quarterly Journal of Economics* **108(1)**: pp. 137–156.

KRISTENSEN, D. (2009): "Uniform convergence rates of kernel estimators with heterogeneous dependent data." *Econometric Theory* **25(5)**: pp. 1433–1445.

KRISTENSEN, D. & Y. SHIN (2012): "Estimation of dynamic models with nonparametric simulated maximum likelihood." *Journal of Econometrics* **167(1)**: pp. 76–94.

KUKACKA, J. & J. BARUNIK (2017): "Estimation of financial agent-based models with simulated maximum likelihood." *Journal of Economic Dynamics and Control* **85**: pp. 21–45.

KUKACKA, J. & S. SACHT (2023): "Estimation of heuristic switching in behavioral macroeconomic models." *Journal of Economic Dynamics and Control* **146**: p. 104585.

KUKACKA, J. & E. ZILA (2024): "Wealth, cost, and misperception: Empirical estimation of three interaction channels in a financial-macroeconomic agent-based model." *Available at SSRN 4709540* .

LAMPERTI, F., A. ROVENTINI, & A. SANI (2018): "Agent-based model calibration using machine learning surrogates." *Journal of Economic Dynamics and Control* **90**: pp. 366–389.

LEE, B.-S. & B. F. INGRAM (1991): "Simulation estimation of time-series models." *Journal of Econometrics* **47(2-3)**: pp. 197–205.

LUX, T. (2018): "Estimation of agent-based models using sequential monte carlo methods." *Journal of Economic Dynamics and Control* **91**: pp. 391–408.

LUX, T. (2022): "Bayesian estimation of agent-based models via adaptive particle markov chain monte carlo." *Computational Economics* **60(2)**: pp. 451–477.

LUX, T. (2023a): "Approximate bayesian inference for agent-based models in economics: a case study." *Studies in Nonlinear Dynamics & Econometrics* **27(4)**: pp. 423–447.

LUX, T. (2023b): "Sequential monte carlo squared for agent-based models." In "Artificial Intelligence, Learning and Computation in Economics and Finance," pp. 59–69. Springer.

LUX, T. & R. C. ZWINKELS (2018): "Empirical validation of agent-based models." In "Handbook of computational economics," volume 4, pp. 437–488. Elsevier.

MARJORAM, P., J. MOLITOR, V. PLAGNOL, & S. TAVARÉ (2003): "Markov chain monte carlo without likelihoods." *Proceedings of the National Academy of Sciences* **100(26)**: pp. 15324–15328.

MCFADDEN, D. (1989): "A method of simulated moments for estimation of discrete response models without numerical integration." *Econometrica: Journal of the Econometric Society* pp. 995–1026.

MIAH, M., A. RAHMAN *et al.* (2016): "Modelling volatility of daily stock returns: Is garch (1, 1) enough." *American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)* **18(1)**: pp. 29–39.

MURPHY, K. P. (2012): *Machine learning: a probabilistic perspective.* MIT press.

MUTH, J. F. (1961): "Rational expectations and the theory of price movements." *Econometrica: journal of the Econometric Society* pp. 315–335.

NAESSETH, C. A., F. LINDSTEN, T. B. SCHÖN *et al.* (2019): "Elements of sequential monte carlo." *Foundations and Trends® in Machine Learning* **12(3)**: pp. 307–392.

PAKES, A. & D. POLLARD (1989): "Simulation and the asymptotics of optimization estimators." *Econometrica: Journal of the Econometric Society* pp. 1027–1057.

PLATT, D. (2020): "A comparison of economic agent-based model calibration methods." *Journal of Economic Dynamics and Control* **113**: p. 103859.

PLATT, D. (2022): "Bayesian estimation of economic simulation models using neural networks." *Computational Economics* **59(2)**: pp. 599–650.

RUBIN, D. B. (1984): "Bayesianly justifiable and relevant frequency calculations for the applied statistician." *The Annals of Statistics* pp. 1151–1172.

Sathe, A. M. & N. Upadhye (2022): "Estimation of the parameters of symmetric stable arma and arma–garch models." *Journal of Applied Statistics* **49(11)**: pp. 2964–2980.

Sisson, S. A., Y. Fan, & M. M. Tanaka (2007): "Sequential monte carlo without likelihoods." *Proceedings of the National Academy of Sciences* **104(6)**: pp. 1760–1765.

Souza, R. & A. Neto (1996): "A bootstrap simulation study in arma (p, q) structures." *Journal of Forecasting* **15(4)**: pp. 343–353.

Tavaré, S., D. J. Balding, R. C. Griffiths, & P. Donnelly (1997): "Inferring coalescence times from dna sequence data." *Genetics* **145(2)**: pp. 505–518.

Welch, H. (1983): "Distributed recovery block performance in a real-time control loop." In "Proceedings, IEEE Real-Time Systems Symposium," pp. 268–276.

Zeeman, E. C. (1974): "On the unstable behaviour of stock exchanges." *Journal of mathematical economics* **1(1)**: pp. 39–49.

Zhang, J., Q. Zhang, Y. Li, & Q. Wang (2023): "Sequential bayesian inference for agent-based models with application to the chinese business cycle." *Economic Modelling* **126**: p. 106381.

Zila, E. & J. Kukacka (2023): "Moment set selection for the smm using simple machine learning." *Journal of Economic Behavior & Organization* **212**: pp. 366–391.

Appendix A

Codebase

The whole source code with script files, implementation of individual methods, cached results, and Jupyter Notebooks analyzing the results is available inside the archived GitHub repository:

`https://github.com/okarlicek/DiplomaThesis`.