DEGREE PROJECT IN MATHEMATICS,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2017

On-Line Market Microstructure Prediction Using Hidden Markov Models

MÅNS TILLMAN

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ENGINEERING SCIENCES

On-Line Market Microstructure Prediction Using Hidden Markov Models

MÅNS TILLMAN

Degree Projects in Mathematical Statistics (30 ECTS credits)
Degree Programme in Mathematics (120 credits)
KTH Royal Institute of Technology year 2017
Supervisor at Scila AB: Lars-Ivar Sellberg
Supervisor at KTH: Jimmy Olsson
Examiner at KTH: Jimmy Olsson

TRITA-MAT-E 2017:29
ISRN-KTH/MAT/E--17/29--SE

Royal Institute of Technology
School of Engineering Sciences
KTH SCI
SE-100 44 Stockholm, Sweden
URL: www.kth.se/sci

Abstract

Over the last decades, financial markets have undergone dramatic changes. With the advent of the arbitrage pricing theory, along with new technology, markets have become more efficient. In particular, the new high-frequency markets, with algorithmic trading operating on micro-second level, make it possible to translate "information" into price almost instantaneously. Such phenomena are studied in the field of *market microstructure theory*, which aims to explain and predict them.

In this thesis, we model the dynamics of high frequency markets using non-linear *hidden Markov models* (HMMs). Such models feature an intuitive separation between observations and dynamics, and are therefore highly convenient tools in financial settings, where they allow a precise application of domain knowledge. HMMs can be formulated based on only a few parameters, yet their inherently dynamic nature can be used to capture well-known intra-day seasonality effects that many other models fail to explain.

Due to recent breakthroughs in Monte Carlo methods, HMMs can now be efficiently estimated in real-time. In this thesis, we develop a holistic framework for performing both real-time inference and learning of HMMs, by combining several particle-based methods. Within this framework, we also provide methods for making accurate predictions from the model, as well as methods for assessing the model itself.

In this framework, a sequential Monte Carlo bootstrap filter is adopted to make on-line inference and predictions. Coupled with a backward smoothing filter, this provides a forward filtering/backward smoothing scheme. This is then used in the sequential Monte Carlo expectation-maximization algorithm for finding the optimal hyper-parameters for the model.

To design an HMM specifically for capturing information translation, we adopt the observable *volume imbalance* into a dynamic setting. Volume imbalance has previously been used in market microstructure theory to study, for example, *price impact*. Through careful selection of key model assumptions, we define a slightly modified observable as a process that we call *scaled volume imbalance*. The outcomes of this process retain the key features of volume imbalance (that is, its relationship to price impact and information), and allows an efficient evaluation of the framework, while providing a promising platform for future studies. This is demonstrated through a test on actual financial trading data, where we obtain high-performance predictions. Our results demonstrate that the proposed framework can successfully be applied to the field of market microstructure.

Sekventiell mikrostrukturprediktering med dolda Markovmodeller

Sammanfattning

Under de senaste decennierna har det gjorts stora framsteg inom finansiell teori för kapitalmarknader. Formuleringen av arbitrageteori medförde möjligheten att konsekvent kunna prissätta finansiella instrument. Men i en tid då högfrekvenshandel numera är standard, har omsättningen av information i pris börjat ske i allt snabbare takt. För att studera dessa fenomen; prispåverkan och informationsomsättning, har mikrostrukturteorin vuxit fram.

I den här uppsatsen studerar vi mikrostruktur med hjälp av en dynamisk modell. Historiskt sett har mikrostrukturteorin fokuserat på statiska modeller men med hjälp av icke-linjära dolda Markovmodeller (HMM:er) utökar vi detta till den dynamiska domänen.

HMM:er kommer med en naturlig uppdelning mellan observation och dynamik, och är utformade på ett sådant sätt att vi kan dra nytta av domänspecifik kunskap. Genom att formulera lämpliga nyckelantaganden baserade på traditionell mikrostrukturteori specificerar vi en modell—med endast ett fåtal parametrar—som klarar av att beskriva de välkända säsongsbeteenden som statiska modeller inte klarar av.

Tack vare nya genombrott inom Monte Carlo-metoder finns det nu kraftfulla verktyg att tillgå för att utföra optimal filtrering med HMM:er i realtid. Vi applicerar ett så kallat *bootstrap filter* för att sekventiellt filtrera fram tillståndet för modellen och prediktera framtida tillstånd. Tillsammans med tekniken *backward smoothing* estimerar vi den posteriora simultana fördelningen för varje handelsdag. Denna används sedan för statistisk inlärning av våra hyperparametrar via en sekventiell Monte Carlo Expectation Maximization-algoritm.

För att formulera en modell som beskriver omsättningen av information, väljer vi att utgå ifrån *volume imbalance*, som ofta används för att studera prispåverkan. Vi definierar den relaterade observerbara storheten *scaled volume imbalance* som syftar till att bibehålla kopplingen till prispåverkan men även går att modellera med en dynamisk process som passar in i ramverket för HMM:er. Vi visar även hur man inom detta ramverk kan utvärdera HMM:er i allmänhet, samt genomför denna analys för vår modell i synnerhet. Modellen testas mot finansiell handelsdata för både terminskontrakt och aktier och visar i bägge fall god predikteringsförmåga.

Acknowledgements

I am most grateful to Lars-Ivar Sellberg and Scila AB for introducing me to their contacts at Deutsche Börse AG and for sponsoring my trip to Frankfurt for extracting the financial data required for this thesis. Many thanks also for their expert advice on market regulation, which I have tried to incorporate into this thesis to make it highly interesting also from a regulatory point-of-view.

I would like to extend my genuine thanks to Carl-Frederik Scharffenorth and Deutsche Börse AG for providing me with data for this thesis. Without this data, this thesis would not have been possible.

Furthermore, I want to express my gratitude to my supervisor Jimmy Olsson at the Royal Institute of Technology (KTH) for his invaluable comments and guidance.

Last, I would like to thank Emma Tegling at the Royal Institute of Technology (KTH) for her assistance with proof-reading and advice on scientific writing.

Contents

Contents **vi**

1 Introduction **1**
1.1 Purpose . . . 1
1.2 Thesis outline . . . 2
1.3 Delimitations . . . 2
1.4 Notation . . . 3

2 Background and Preliminaries **5**
2.1 Market microstructure theory . . . 5
2.2 Statistical definitions . . . 8
2.3 Monte Carlo methods . . . 13

3 Model **25**
3.1 The scaled volume imbalance . . . 25
3.2 Making assumptions . . . 26
3.3 Defining the model . . . 29

4 Method **33**
4.1 Framework . . . 33
4.2 Implementation . . . 36

5 Results **39**
5.1 Learning the hyperparameter . . . 39
5.2 Parameter inference . . . 42
5.3 Posterior predictive checks . . . 43

6 Discussion **49**
6.1 Notes on the framework . . . 49
6.2 Data handling . . . 50
6.3 Notes on the scaled volume imbalance . . . 50
6.4 Intra-day changes . . . 51
6.5 Sampling parameters . . . 51
6.6 Information carried by trades . . . 52

7 Conclusions and Future work **55**
7.1 Conclusions 55
7.2 Future work 56

8 Appendix **57**
8.1 Proofs 57

Bibliography **59**

Chapter 1

Introduction

During the last couple of decades, financial markets have undergone dramatic changes. With the advent of the Arbitrage Pricing Theory and new technology, markets have become more efficient. Through algorithmic trading, operating on micro-second level, the new high-frequency markets make it possible to translate "information" into price almost instantaneously. But what do we mean by the concept of information and how can we quantify and measure it?

Those are topics which are being studied intensively by many research teams at this very moment. The critical component when studying information is a concept called *price impact*. This has been modelled in numerous different ways, but almost always with the common denominator that the models are static. This leads to a number of unwelcome side-effects, such as failure to explain intra-day seasonality.

With this thesis, we aim to provide a framework for modelling and testing market microstructure phenomena, like the the price impact, in a dynamic Bayesian setting, using non-linear hidden Markov models. In particular, we define the *scaled volume imbalance*, which is closely related to price impact, and develop a model for successfully tracking this quantity using the provided framework.

1.1 Purpose

This thesis has two main purposes. The first purpose is to cast standard market microstructure theory into a Monte Carlo framework by defining a hidden Markov model for capturing and predicting realization of market information. In order to justify this, we will provide insight to the adequacy of using hidden Markov models in a financial context—with a focus on high-frequency markets—through thorough discussions on model details and key assumptions. Based on these insights, we will then define our hidden Markov model.

The second purpose is to show how recent particle-based Monte Carlo methods can be combined into a holistic framework for studying such hidden Markov models. We will define applicable forward and backward particle filters and discuss how they together can be used to solve both the inference problem and the learning problem in

a super-efficient way. In particular, we will show how these methods can be applied to the hidden Markov model for making predictions, assessing model performance and spotting market anomalies.

1.2 Thesis outline

In Chapter 2 we present all relevant theory needed for this thesis. The basics of market microstructure theory are explained and a number of useful Monte Carlo methods are derived. All algorithms are given in detail and proofs are provided or outlined.

In Chapter 3 we define the scaled volume imbalance and develop a suitable dynamic model to describe this quantity. We discuss all relevant assumptions and benefits with this model, as well as its associated parameters, thoroughly.

In Chapter 4 we describe how sequential Monte Carlo methods can be used for parameter and state inference in hidden Markov models, such as the one we have defined for the scaled volume imbalance. This framework encompasses everything from making inference about state parameters and making predictions, to learning hyperparameters and providing methods for justifying the model.

In Chapter 5 we use the framework developed in Chapter 4 to study the model defined in Chapter 3. The model is put to test using data obtained from Deutsche Börse AG for trading in stock equity and futures contract instruments during a period of two weeks in February, 2016. All relevant results are provided and explained. In the remaining part of this thesis strengths, weaknesses, possible room for improvement and further extensions to the proposed framework are discussed in the light of the results.

1.3 Delimitations

In this thesis we will define the scaled volume imbalance such that it will have a close connection to price impact—similarly to the standard volume imbalance. However, actually defining this relationship to price impact is considered out-of-scope.

The framework that we will develop in this thesis does not include any sensitivity analysis in relation to the likelihood functions. This is considered superfluous in the context of the other analyses. Also, the study of likelihood sensitivity is not critical when assessing a single model.

When modelling the scaled volume imbalance we will not investigate the possibility of correlated parameter movements. Any such correlation is considered beyond the scope of this thesis.

The sample interval length Δt will not be considered part of the hyperparameter set θ. We will examine the impact on predictions for changes in Δt, but we will not set out to find an optimal value for this.

Symbol	Description
HMM	Hidden Markov model
JSD	Joint smoothing density
MC	Monte Carlo
SMC	Sequential Monte Carlo
$\xi^i_{1:t}$	The trajectory from time 1 to t for particle i
$\{\xi^i\}_{i=1}^N$	A set of N particle values
$\{\xi^i, w^i\}_{i=1}^N$	A set of N weighted particles, approximating the density of x
X_t	The latent random variables of an HMM for a certain timestep t
x_t	The outcome of latent variables of an HMM for a certain timestep t
Y_t	The random variables associated with outcomes of an HMM for a certain timestep t
y_t	The observed outcomes of an HMM for a certain timestep t
$\phi_{t:t'\|T}$	The JSD from time t to t', given observations up to time T
ϕ_t^N	The N-particle marginal filter density at time t
$\phi_{t:t'\|T}^N$	The N-particle JSD approximation from time t to t', given observations up to time T
$\hat{\phi}_{t+1\|t}^N$	The N-particle predictive density, given the observations up to time t
$\hat{\varphi}_{MC}^N$	An N-particle MC estimator of $\mathbb{E}[\varphi(X)]$

Table 1.1. Common notation used in this thesis.

1.4 Notation

We will write sequences of values in the short-hand notation $x_{0:t} \overset{\text{def}}{=} \{x_0, \ldots, x_t\}$. Probability densities associated with distributions are in general denoted by p throughout this thesis. Further, we will use the notation $p(x_k)$ to denote the probability that X_k at a certain time k assumes the value x_k under p, that is $p(x_k) \overset{\text{def}}{=} \mathbb{P}(X_k = x_k)$. Similarly, for a conditional distribution on some random variable Y_k we will write $p(x_k \mid y_k) \overset{\text{def}}{=} p(x_k \mid Y_k = y_k)$.

Throughout this thesis we will consider distributions of x_k conditional on sequences of historical outcomes of random variables $\{X_0 = x_0, \ldots, X_{k-1} = x_{k-1}\}$. Using the above notation, the probability density for this conditional distribution simplifies to $p(x_k \mid x_{0:k-1})$.

Dependency on any hyperparameters θ in distributions is indicated by a subscript. Thus, for a density p that depends on θ we will write $p_\theta(x)$.

Together with this we employ the notation defined in Table 1.1, which is adopted from contemporary Monte Carlo literature.

Chapter 2

Background and Preliminaries

In this chapter we will look at earlier relevant research and tools that we will make use of in this study.

2.1 Market microstructure theory

The primary driving force in trading is *information*. Unless you are completely indifferent to the outcome of your investment, you will make the decision to trade based on some kind of information that is available to you. By the term "information" we do not restrict ourselves to specific news events related the particular asset itself, but include all information that is of interest when deciding on trading strategies. This includes everything from the supply of the asset and the state of the entity supplying the asset, to the collective buying power in the markets, as well as the full utilities and strategies of each and every trading participant. These different sources of information can by divided into two different categories; *macroscopic* and *microscopic*. Macroscopic information is what is usually known as the *fundamentals* for the asset. This is a slow process. Microscopic information, on the other hand, is the information held by all market participants—the *traders*—and is a process that can be changing very rapidly. In this thesis we will focus on the latter, which is studied in the field of *market microstructure theory*.

2.1.1 Background

In the early 1990's computers were starting to make their way into financial markets. This technological change lead to new conditions for the markets' participants as information became more readily available and the process of placing orders was made easier. From the exchanges' perspective the new technology enabled new ways of collecting and keeping records of the trading activity. This record-keeping was also enforced by the introduction of series of new regulatory laws, which were a consequence of an increased demand for transparency.

Consequentially, researchers suddenly had transaction data of a much higher quality than had ever been seen before at their disposal. With this, new opportunities to analyse trading dynamics and to describe what was actually going on in the markets came into existence. Using this data from the computer-powered financial exchanges, several groups of researchers set out to identify the mechanics of financial trading from a mathematical point-of-view. This branch of finance is today known as market microstructure theory and center around the driving dynamics of financial trading. A good summary on the foundation of market microstructure theory can be found in the book by the same name by O'Hara [16].

Understanding the dynamics boils down to analysing the behaviour of the traders. Beside detailed modelling of behaviour, market microstructure theory also encompasses everything from optimal trading strategies for reducing transaction costs, to making inference on the amount of informed traders being active in the market at any given point in time. In order to further explore internal dynamics of trading, a new scientific sub-branch called Limit Order Book (LOB) modelling emerged in the early 2000's. Since then, many fascinating articles have been published that accurately explain empirically observed phenomena—such as the concave price impact in relation to volume—in the context of information.

In LOB modelling the complete order flow is generally assumed to carry information. This means that all actions carried out by all traders together define the preconditions for trading. Despite proposals of numerous models, we are yet to see how to utilize the full width of information in circulation.

2.1.2 The volume imbalance

In [18] the concept of *price impact* is studied as an effect of demand fluctuations. The *volume imbalance* Ω is defined as

$$\Omega(t) \stackrel{\text{def}}{=} Q_B - Q_S = \sum_{i=1}^{N} q_i a_i, \tag{2.1}$$

where Q_B are the buyer-initiated transactions and Q_S are the seller-initiated transactions. Further, q denotes the volume for each trade and a the sign of the trade. This quantity is used to act as a proxy for the demand fluctuations in the market. A distinct relationship between the volume imbalance and the price impact is established through analysing a large number of US traded stocks spanning the period of 1994-1995.

In the paper, Ω is presented purely as observations and used to explain price impact. Therefore, no assumptions are made in relation to how the traded volume is generated. In order to enable predictions of future price impact, the quantity itself must be modelled in some way.

2.1.3 Sources of information

The actions in the order book that are assumed to carry information (and thereby possible sources of information) are the following [3, 6]:

(A) To place a limit order

(B) To cancel a limit order

(C) To place a market order

As the terminology differs slightly across platforms, we will walk through what we mean by each of these. Beginning from the top, Action (A) means that a trader enters an order to sell (or buy) q contracts for asset I at a price p. This *limit order* goes into the orderbook for asset I and waits there for someone to accept this offer. The removal of such an offer, is action (B).

The last action, (C), means that a trader has found an existing offer that they are willing to accept. They place a *market order* to hit this active limit order. The result of this is that a *trade* is executed. A trade is an event where one trader pays fiat currency to another trader in exchange for a certain number of contracts for a financial asset.

2.1.4 Concave price impact

Price impact has been shown in many papers to be *concave* with respect to the volume of a trade. In the paper by *Plerou et al* [18] (where the volume imbalance was proposed) the functional form of this relationship is determined. In particular the power-law

$$\Delta p \sim \Omega^{\beta}$$

is studied and applied successfully with values of β ranging from 1/3 up to 1. Here Δp is the expected price impact over the sampling time period Δt, studied in terms of Ω. The exponent is shown to increase with Δt. This would suggest that the number of trades could be playing a role here as well—not only the aggregated volumes. Such scaling effects are seen in many areas of market microstructure theory.

2.1.5 Trade-by-trade concavity

In [12] a similar approach is taken, also finding a power-law relation to price impact. However, in their paper, the price impact of volume is studied in the context of *individual trades*, rather than to an aggregated volume imbalance over time. They express the price impact of a single trade, denoted Δp, in terms of the trade's volume q, and sign a as

$$\Delta p = a\frac{q^{\beta}}{C}, \tag{2.2}$$

where C is a liquidity constant. They find that $\beta = 1/2$ generally represents price impact in high-capitalization stocks well. The approach to study price impact trade-by-trade is successful, since it leads to slightly more consistent results.

2.2 Statistical definitions

In this chapter we will define relevant mathematical properties and concepts that will be frequently used throughout this thesis.

2.2.1 Memorylessness

Any probability distribution satisfying the below identity is said to be *memoryless*.

$$\mathbb{P}\left(X > t + s \mid X > t\right) = \mathbb{P}\left(X > s\right) \tag{2.3}$$

One way of explaining this property is to consider waiting times. Assume that we have three trades that arrive at times t_1, t_2 and t_3. The waiting times are then defined as $t_2 - t_1$ and $t_3 - t_2$. If those waiting times are *independent* the trade flow is said to be memoryless. To find out how suitable the assumption of memorylessness is, it is often quite easy to imagine what causal implication would be caused by memorylessness. This property is frequently used in LOB modelling.

In the continuous case, the only distribution having this property is the Exponential distribution.

2.2.2 Markov chain

A *Markov chain* is a random process that makes discrete transitions in state-space. Given a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ with filtration $\{\mathcal{F}_t, t = 0, 1, \ldots\}$, the stochastic process $\{X_t, t = 0, 1, \ldots\}$ adapted to the filtration is called a Markov chain if it carries the below property.

$$\mathbb{P}(X_{k+1} = x_{k+1} \mid X_0 = x_0, \ldots, X_k = x_k) = \mathbb{P}(X_{k+1} = x_{k+1} \mid X_k = x_k) \tag{2.4}$$

This property is called the *Markov property*. The interpretation is that the transition probabilities only depend on the present state—which could be thought of as a type of memorylessness. Markov chains are well suited for making inference about dynamic systems.

2.2.3 The inhomogeneous Poisson process

An *inhomogeneous Poisson process* is a counting process defined as the total number of events up to point t in time. The difference $N(t + \Delta t) - N(t)$ is a Poisson-

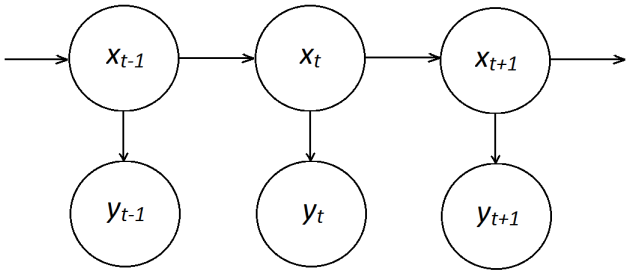


Figure 2.1. A graphical representation of a hidden Markov model. The latent variables x form a Markov chain and the outcomes y are conditionally independent.

distributed random variable with parameter λ_t, which is defined by

$$\lambda_t = \int_t^{t+\Delta t} \lambda(s) ds,$$

where $\lambda(s)$ is the instantaneous value of the intensity.

In this thesis we will only be considering λ as a parameter in a Markov chain evolving on an equidistant grid defined by Δt. Therefore, $\lambda(s)$ will be a piece-wise constant function. We use t to index the λ parameter accordingly. An inhomogeneous Poisson process defined this way carries the memorylessness property in that the inter-arrival times are exponentially distributed with the parameter λ_t.

In most of the current LOB modelling literature only homogeneous Poisson processes are used. This means that the parameter λ_t is constant, hence not dependent on t. By extending the parameter to be a function of time we can, for example, successfully address the peculiarity called *diurnality*, which is the increased trading at the beginning and the end of the trading day. A thorough discussion on the use of Poisson processes in econometrics can be found in [2].

2.2.4 The hidden Markov model

A *hidden Markov model* (HMM) describes the evolution of a system consisting of a set of *latent variables* x. The word "latent" refers to the notion of these system variables being impossible to observe directly. Instead, they manifest through a series of observations y. The latent variables form a *Markov chain* and the observations are conditionally independent, given the latent variables (see Figure 2.1). As we can see, at every point in time t_k the process will have the state x_{t_k} and yield the observable outcome y_{t_k}.

The HMM is defined by the underlying Markov chain of the latent variables, along with the relationship between the outcomes and the latent variables. The Markov chain in turn is defined by the evolution probabilities of the latent variables, called the *transition kernel*, with the associated transition density q, and

the initial distribution χ of the variables. The observational relationship is defined by the *observation density* p, which is the conditional distribution of $y_t \mid x_t$. The densities depend on a set of hyperparameters θ and can take any possible shape, hence allowing strongly non-linear behaviour. This is summarized by the following relationships that together define the Markov chain.

Definition 1 (Hidden Markov model)**.** A model with latent variables x forming a Markov chain, with associated observable variables y that are conditionally independent given the latent variables, is called a hidden Markov model if it has a transition kernel, observation density and initial distribution of the following form

$$\begin{aligned} y_t \mid x_t &\sim p_\theta(y_t \mid x_t) \\ x_{t+1} \mid x_t &\sim q_\theta(x_{t+1} \mid x_t) \\ x_0 &\sim \chi(x_0) \end{aligned}$$

2.2.5 Maximum likelihood estimation

A technique often used is statistics is *maximum likelihood estimation*. Assume that a sequence of outcomes y were generated by a function of some parameter θ. In order to formulate a good point estimator for θ, we consider the likelihood function for θ, given the outcomes y. The likelihood function is written as

$$\mathcal{L}(\theta; y) = p_\theta(y),$$

where $p_\theta(y)$ is the joint probability for the sequence of outcomes y for a specific θ. Using this definition, the *maximum likelihood estimator* (MLE) is defined as

$$\hat{\theta} = \arg\max_\theta \mathcal{L}(\theta; y).$$

This gives the point estimator of θ for which we obtain the highest likelihood of observing the specific sequence of outcomes y. From a Bayesian perspective, the MLE coincides with the maximum a posteriori estimator of θ when a uniform prior is assumed, i.e. when no prior information is held about the distribution of the (in this case) random variable θ.

2.2.6 The learning problem

The task of defining a mathematical model which can accurately reflect a system is in the field of statistics called the *learning problem*. This includes everything from making the choice whether to use a parametric or non-parametric model to determining the model's functional form.

The parameters by which the model is parametrized, are called *hyperparameters* and are denoted by θ. In this thesis we will address the learning problem by computing the MLE of the hyperparameters. However, for doing so, there are still some problems that we have to address. For example, the likelihood function is, unfortunately, in general not analytically tractable.

2.2.7 Expectation-maximization

For a hidden Markov model approach, the likelihood function generally becomes intractable due to the nature of the latent variables x. It should be noted, though, that this is not the case in, for example, systems with linear-dynamic variables having Gaussian observation densities. In [1] a technique called *data augmentation* i proposed to be used for addressing this intractability. The trick is to augment the set of observed outcomes $y_{0:T}$ with the unobservable outcomes $x_{0:T}$ (the state). The resulting set $\{x_{0:T}, y_{0:T}\}$ is called the *complete data*.

By using the complete data it is possible to express the likelihood in terms of the joint density by the relation

$$p_\theta(y_{0:T}) = \frac{p_\theta(x_{0:T}, y_{0:T})}{p_\theta(x_{0:T} \mid y_{0:T})}. \tag{2.5}$$

This construction is then used to formulate an *expectation-maximization* (EM) algorithm for maximum likelihood estimation in scenarios with incomplete data. The algorithm consists of the following two steps—(E) and (M)—that are repeated iteratively. The EM algorithm is summarized in Algorithm 1.

Algorithm 1: The EM Algorithm

Data: Initial guess θ'
Result: MLE $\hat{\theta}$
while *Stopping condition not met* **do**
  (E) Compute $\mathcal{Q}(\theta, \theta') = \mathbb{E}_{\theta'}\left[\log\left(p_\theta(x_{0:T}, y_{0:T})\right) \mid y_{0:T}\right]$
  (M) Update $\theta' = \arg\max_{\theta \in \Theta} \mathcal{Q}(\theta, \theta')$
end
Set $\hat{\theta} = \theta'$

After the stopping condition has been met, the final θ' can be considered optimal and thereby the learning problem is solved. We have outlined the proof for the EM algorithm in Section 8.1.1 of the Appendix.

2.2.8 The state inference problem

In addition to model learning, we will in this thesis also address the *state inference problem*. There are three types of state inference problems. In the HMM setting, these problems are all in some way concerned with finding the posterior distribution $p_\theta(x \mid y)$, which is the state probability density, given the set of observed outcomes y. The difference between the three problems can be expressed in terms of the time period the inference is targeting. See the table below for the the target density associated with each problem.

Problem	**Target density**
Smoothing problem	$p_\theta(x_{0:t} \mid y_{0:t})$
Filtering problem	$p_\theta(x_t \mid y_{0:t})$
Prediction problem	$p_\theta(x_{t+1} \mid y_{0:t})$

It should be noted that, in general, the smoothing problem does not necessarily concern the whole time-range from time 0, but is rather inference about any state prior to t. In the same way the prediction problem, in general, refers to any inference after t. Because of the different nature of the problems, they will be tackled using different methods. However, by using the Monte Carlo framework it will be possibly to do this in a synergistic way. This will be shown later in this thesis as we will touch upon each of these problems in some way.

2.2.9 Single model approach

After the model has been defined, we are ready to evaluate, or *assess*, the model. In a Bayesian setting, a model is generally not assessed on its own, but in the context of one or more other models. It is, however, possible to justify a single model from a Bayesian perspective as well. Even though the framework applied in this thesis is not properly Bayesian, the approach outlined below can still be successfully used for assessing HMMs.

In the 80's a number of papers were published addressing how to properly assess Bayesian models (see e.g. [19] and [20]). We have listed the three key features that can be used for justifying a single model below.

1. Sensitivity to the prior and the likelihood

2. Legitimacy of the posterior

3. Fitness to data

The first item conveys the importance of checking the posterior distribution by analysing how it is affected by changes in its two sub-components; the prior and the likelihood. The topic concerned with analyses of this kind is known as *robust Bayesian analysis*, or *Bayesian sensitivity analysis*. In the context of this thesis, this translates into studying how sensitive the posterior is to changes in θ. Regarding the second item, this is often done by examining the resulting posterior distribution to see that its associated properties are intuitively correct and satisfy the requirements. For example, does the support of the posterior cover the observable space Y? Does the skewness correspond to what we expect it to be? Does the number of modes correspond to what we expect it to be? And so on.

Still, regardless of all else, the most important feature of the selected model is its fitness to actual data. If the model does not characterize the data appropriately, the model cannot be justified.

2.2.10 Posterior predictive checks

To address how to assess the fitness to data, a *posterior predictive check* is proposed in [20]. To perform this, a test statistic T for the observed outcome y_{t+1} is compared to that of a replicated observation y_{t+1}^{rep}, given the history of observed outcomes $y_{1:t}$. The models treated in his original papers are all static, meaning that the distribution at time $t+1$ is assumed to be the same as that of time t. This can, however, easily be extended to the dynamic setting used in this thesis.

By the construction of the test statistics it is possible to define what is called the *posterior predictive p-value*

$$p(y_{t+1}) = \mathbb{P}\left(T(y_{t+1}) \geq T(y_{t+1}^{rep}) \mid y_{1:t}, \theta\right). \tag{2.6}$$

Note that the expression above will average p over the whole posterior. Thus, this is basically a way of measuring the tail probability for some test statistic, given the realized outcome.

The possibility of interpreting this entity as the standard p-value, to be used in the same way as in the frequentist setting, has been discussed a lot in the literature. In essence, by defining T in such a way that the properties associated with the p-values are known, those properties can be used to formulate hypothesis tests.

2.3 Monte Carlo methods

In this section we will go through a number of sophisticated techniques to use for learning and inference in hidden Markov models (see Definition 1), called *Monte Carlo methods*. Proofs are provided where it is practical and in other places brief outlines of the derivations of proofs are given.

For a basic walkthrough of established Monte Carlo methods see, for example, [4] or [14] for good monographs on the subject. For more in-depth treatment of the methods see, for example, the tutorial [9]. Also, for more recent convergence results in some of the more advanced methods see, for example, [7, 17].

2.3.1 Background

The modern development of Monte Carlo methods started over sixty years ago by Metropolis and Ulam [15]. This paper devised a method to solve integration of high-dimensional physical differential equations by using randomly generated numbers. Since then, the methods have advanced enormously and now covers a wide range of problems, and they are currently used frequently in everything from molecular biology to voice recognition and computer vision.

The way Monte Carlo methods work, is that by locating the part of the high-dimensional space that has *high importance*, only a fraction of the space needs to be covered to accurately approximate the integral. Therefore, by generating a sample of high importance from a random distribution, the target distribution can be approximated via empirical probability density distributions. Empirical densities

are discrete, but instead of handling this discrete property by employing an equidistant grid, the Monte Carlo methods use a finite number of *particles*. By having these particles approximate independent draws from the target density, we ensure that every point holds a lot of information. This way, it is possible to construct algorithms that are more computationally efficient than using the standard numeric integration.

Further, even more important, is the capability of approximating sequences of target distributions well that the Monte Carlo methods have. By sequencing in the time-dimension, this trait can be used for solving high-dimensional problems over time. This is not limited to solving static systems, but was quickly adopted to handle dynamic systems as well. In particular, using MC methods has proven to be very successful in making inference about HMMs. The reason for this is that making inference about a state-space model is equivalent to computing sequences of posterior distributions. Given the usefulness of HMMs for making inference about systems that evolve over time, a new type of models capable of making on-line inference of such processes were developed. These are called *sequential Monte Carlo* (SMC) methods. The foundation for SMC methods can be found in the book [8]. For a well-written and easily accessible introduction to SMC [9] is recommended. By the use of SMC methods, it is possible to continuously update the inference as new observations are made available.

SMC algorithms are usually utilized to compute the filtered marginal posterior, rather than the full joint posterior. The reason behind this is that the SMC methods are very good at telling where we are at the moment of the new observation, but have trouble describing the bigger picture. This primarily due to a side-effect called *path degeneracy*, which we will discuss further later in this section.

In recent days, there has been a spiking interest in improved *backward smoothing* algorithms, which in combination with a regular forward SMC, can recover the non-degenerate joint posterior distribution. An overview and comparison of such algorithms can by found in [7].

2.3.2 Monte Carlo Integration

Before going into any more detail, we will first go through the intuition of Monte Carlo integration. Assume that we want to compute an integral over some high-dimensional space X. This problem often arises in the context of computing an expected value

$$\mathbb{E}_p\left[\varphi(X)\right] = \int_{\mathsf{X}} \varphi(X)p(X)dX, \tag{2.7}$$

where X is a random variable on some probability space $(\mathsf{X}, \mathcal{X}, p)$.

The way Monte Carlo methods deal with this is that instead of discretizing the whole space X and integrating over every single point, the points to use for integration are randomly drawn from the target density p.

In most cases, however, the target density will be intractable and hence not possible to sample from. In order to address this, several sophisticated methods have been developed. Generally, a proposal kernel that emphasizes the important regions of the integral is used. This keeps the complexity of the problem down considerably compared to standard numeric integration.

2.3.3 The MC sampler

Any estimator for the integral (2.7) is generally called a *Monte Carlo estimator*. For the purpose of this thesis, we will require the Monte Carlo estimator to carry certain properties to guarantee its usefulness. This is summarized in the definition below.

Definition 2 (MC estimator)**.** An MC estimator $\hat{\varphi}^N_{MC}$ is an estimator, that for any random variable X on a probability space $(\mathsf{X}, \mathcal{X}, p)$ and test function $\varphi : X \to \mathbb{R}$ has the following properties

[P1] (Almost sure convergence) $\hat{\varphi}^N_{MC} \xrightarrow{\text{a.s.}} \mathbb{E}_p[\varphi(X)], \quad N \to \infty$

[P2] (Follows CLT) $\frac{\sqrt{N}(\hat{\varphi}^N_{MC} - \mathbb{E}_p[\varphi(X)])}{\sigma_\varphi} \xrightarrow{\text{D}} \mathcal{N}(0,1), \quad N \to \infty$

Following the terminology in [8], we will call the simplest kind of MC estimator a *perfect MC sampler*. This MC sampler is defined as

$$\hat{\varphi}^N_{MC} \overset{\text{def}}{=} \frac{1}{N} \sum_{i=1}^{N} \varphi(\xi^i), \tag{2.8}$$

where the values $\{\xi^i\}_{i=1}^N$ are samples from the probability density p, associated with the measure of the integral. The perfect MC sampler is an MC estimator, as defined in Definition 2. The most interesting thing about this estimator is that it can approximate the exact integral without any knowledge needed about the theoretical distribution for p.

The convergence stated in [P1] follows directly from the *strong law of large numbers*. To prove [P2], let σ^2_φ denote the variance of the random variable $\varphi(X)$. Then the variance of the estimator is given by $\text{Var}(\hat{\varphi}^N_{MC}) = \sigma^2_\varphi / N$. Looking at that expression, we can see that if the variance of $\varphi(X)$ is bounded, then the variance of the estimator is bounded, too. Following this, a *central limit theorem* can be established and the rate of convergence is assured.

In addition to these two properties, this particular MC estimator is also unbiased. However, this is not required for MC estimators in general. The focus is instead set on the efficiency of the estimator.

So far everything is well. However, in many applications of interest, sampling from p is infeasible. For example, in a Bayesian setting, the target distribution is usually a posterior distribution. If the prior distribution is not a conjugate for the likelihood, the posterior is in general analytically intractable and very expensive to

simulate from directly—if possible at all. In order to address this problem, a couple of variations of the perfect MC sampler, that are also MC estimators, have been proposed.

2.3.4 Importance sampling

The key to the development of particle-based Monte Carlo methods lies in the extension of the basic Monte Carlo sampler to the concept of *importance sampling*. The importance sampler is a slight modification to the perfect MC sampler defined in (2.8). By introducing an instrumental density g, we can address the problem of how to sample from the unknown target density p, but still keep the good properties of the estimator.

Definition 3 (IS estimator)**.** An IS estimator $\hat{\varphi}_{IS}^N$ is defined as

$$\hat{\varphi}_{IS}^N \stackrel{\text{def}}{=} \frac{1}{N}\sum_{i=1}^{N} w(\xi^i)\varphi(\xi^i),$$

where

- $w(x) = p(x)/g(x)$,
- $\operatorname{supp}\varphi(x)p(x) \subset \operatorname{supp} g(x)$
- $\{\xi^i, i = 1, \ldots, N\} \sim g$

Using Definition 3 we can then formulate the following lemma

Lemma 1. *The IS estimator is an MC estimator.*

Proof. To show that Lemma 1 holds, we will apply a change of measure

$$\begin{aligned}\mathbb{E}_p\left[\varphi(X)\right] &= \int_{\mathsf{X}} \varphi(X)p(X) = \int_{\mathsf{X}} \varphi(X)\frac{p(X)}{g(X)}g(X)dX \\ &= \int_{\mathsf{X}} \varphi(X)w(X)g(X)dX = \mathbb{E}_g\left[w(X)\varphi(X)\right]\end{aligned}$$

We note that this change of measures is allowed by the definition of g. Since we have already shown that the perfect MC sampler is an MC estimator, we are done. □

2.3.5 Self-normalized importance sampling

In most practical applications, the instrumental density g will only be known up to a normalizing constant c. Thus, we have $g(x) = cg_0(x)$, where $g_0(x)$ is a known density function. In this case the standard IS sampler will not be sufficient for estimating the expectation. We will mitigate this issue by introducing *self-normalization* to the sampler.

Definition 4 (SNIS estimator)**.** A SNIS estimator $\hat{\varphi}^N_{SNIS}$ is defined as

$$\hat{\varphi}^N_{SNIS} \overset{\text{def}}{=} \frac{\sum_{i=1}^N w_0(\xi^i)\varphi(\xi^i)}{\sum_{i=1}^N w_0(\xi^i)},$$

where

- $w_0(x) = p(x)/g_0(x)$,
- $\operatorname{supp} \varphi(x)p(x) \subset \operatorname{supp} g(x)$
- $\{\xi^i, i = 1, \ldots, N\} \sim g$

This method is called *self-normalized importance sampling* and relies on the same construction as the standard IS sampler. The only difference is the introduction of a denominator. We will state the following lemma for capturing the usability of this sampler

Lemma 2. *The SNIS estimator is an MC estimator.*

Proof. To prove Lemma 2, we will start by expanding the expression a bit to re-introduce the old weight function w.

$$\hat{\varphi}^N_{SNIS} = \frac{\sum_{i=1}^N w_0(\xi^i)\varphi(\xi^i)}{\sum_{i=1}^N w_0(\xi^i)} = \frac{\frac{1}{N}\sum_{i=1}^N \frac{cp(\xi^i)\varphi(\xi^i)}{q(\xi^i)}}{\frac{1}{N}\sum_{i=1}^N \frac{cp(\xi^i)}{q(\xi^i)}} = \frac{\frac{1}{N}\sum_{i=1}^N \frac{p(\xi^i)\varphi(\xi^i)}{q(\xi^i)}}{\frac{1}{N}\sum_{i=1}^N \frac{p(\xi^i)}{q(\xi^i)}}$$

Here, we obtained the last expression by identifying the constant c in both enumerator and denominator, thus cancelling each other. To prove the lemma we will look at the enumerator and denominator in the RHS separately. In the enumerator we can see the definition of the standard IS sampler, which is an MC sampler by 1. Looking at the denominator we see that since ξ^i are drawn from g, by the strong law of large numbers, we get that

$$\frac{1}{N}\sum_{i=1}^N \frac{p(\xi^i)}{q(\xi^i)} \xrightarrow{\text{a.s.}} \int_{\mathsf{X}} p(X)dX = 1, \quad N \to \infty,$$

which completes the proof. □

2.3.6 Sequential Monte Carlo

In this thesis we will focus on the SMC methods, which were first proposed in 1969 [10]. These methods are characterized by the possibility to infer the state incrementally through recursive formulas, to incorporate new evidence as it arrives. The recursion is key to make the on-line inference. In this section we consider SMC methods for making inference on general HMMs as defined in Definition 1.

The recursion is derived by splitting the state inference problem into two separate steps, which are called the *measurement update* and the *prediction update*, respectively. The associated formulas are defined below.

$$p(x_{0:t} \mid y_{0:t}) = \frac{p(y_t \mid x_t)p(x_{0:t} \mid y_{0:t-1})}{p(y_t \mid y_{0:t-1})} \tag{2.9}$$

$$p(x_{0:t+1} \mid y_{0:t}) = q(x_{t+1} \mid x_t)p(x_{0:t} \mid y_{0:t}) \tag{2.10}$$

Equation (2.9) is called the *measurement update recursion formula* and as is derived by applying Bayes' theorem to the joint posterior distribution $p(x_{0:t} \mid y_{0:t})$, but only for the most recent y_t. This way we get that $p(x_{0:t} \mid y_{0:t-1}, y_t) = p(y_t \mid x_{0:t}, y_{0:t-1})p(x_{0:t} \mid y_{0:t-1})/p(y_t \mid y_{0:t-1})$. Because of the Markov property of the HMM we realize that the probability for y_t only depends on x_t and therefore the conditioning on $x_{0:t-1}$ and $y_{0:t-1}$ can be dropped from the expression. Here, we identify the first density in the enumerator, $p(y_t \mid x_t)$ as the observation density of an HMM. Further, we note that the function $p(y_t \mid y_{0:t-1})$ in the denominator is the one-step likelihood, which is constant given the observations.

The second equation, (2.10), is called the *time update recursion formula*. To derive this expression we do the separation trick once again but this time for x, i.e. considering $x_{0:t}$ and x_{t+1} separately. Expressing this in term of conditional distributions we obtain $p(x_{0:t}, x_{t+1} \mid y_{0:t}) = p(x_{t+1} \mid x_{0:t}, y_{0:t})p(x_{0:t} \mid y_{0:t})$. Since $y_{0:t}$ does not add information in addition to that contained within $x_{0:t}$, we can be drop it from the conditioning. Doing so, we can identify the first distribution as the transition kernel q in the HMM.

Alternating between inserting the time formula into the measurement formula and vice versa, we can proceed forwards in time sequentially. For each iteration, the only input we need is a new observation y_t.

2.3.7 Particle filters and filter distributions

The recursive method described in the previous section can be adopted to a family of algorithms called *particle filters*. Particle filters are algorithms that use a point-mass approximation $\{\xi_t^i, w_t^i\}_{i=1}^N$ for a probability distribution at time t. In essence, ξ_t^i is an approximated sample from $p(X_t)$ with its associated probability w_t^i. This set of point-mass approximations is called a *weighted particle system*.

For hidden Markov models, the particle filter can be used to make inference on the distribution of the latent variables. As new observations are made available from the true distribution, the algorithm filters the weighted particle system through the new observations. This is done by the recursion formulas defined in (2.9) and (2.10). The marginal filter distribution at time t is denoted by ϕ_t and the weighted N-particle system $\{\xi_t^i, w_t^i\}_{i=1}^N$ approximating this distribution is denoted by ϕ_t^N.

As t increases and new observations are made available, we obtain sequences of weighted particle systems. In the context of ξ_t^i being a particle, the sequence $\xi_{t:t'}^i$ can be thought of as the *particle trajectory* for particle i from time t to t'.

2.3.8 Sequential importance resampling

Using a particle filter for updating the N-particle filter distribution ϕ_t^N when moving from time t to $t+1$ is described in Algorithm 2. This is known as *sequential importance resampling* (SIR). We will outline the derivation of the algorithm below. For a more detailed discussion please see, for example, [9].

Algorithm 2: Sequential importance resampling

Data: y_{t+1}, ϕ_t^N
Result: ϕ_{t+1}^N
for $i = 1, \ldots, N$ **do**
1. Draw $\xi_{t+1}^i \sim q(\xi_{t+1} \mid \xi_{0:t}^i)$
2. Compute $w_{t+1}^i = w_t^i \dfrac{p(y_{t+1} \mid \xi_{t+1}^i) q(\xi_{t+1}^i \mid \xi_t^i)}{g(\xi_{t+1}^i \mid \xi_{0:t}^i, y_{0:t+1})}$

end
3. Normalize weights, s.t. $\sum_{i=1}^N w_{t+1}^i = 1$
4. Set $\phi_{t+1}^N = \{\xi_{t+1}^i, w_{t+1}^i\}_{i=1}^N$

The first step in the algorithm is pretty self-explanatory. It corresponds to a time update recursion, where we only consider the most recent state. This means that to obtain the particle-based equivalent of $p(x_{t+1} \mid y_{0:t})$ we simply mutate our current particles according to the dynamics—which is defined by the transition kernel q.

To derive the second step of the algorithm we first need to change measures to g to define an expression for the weights. Similarly to the standard IS, but this time defining the weights from Definition 3 in the presence of conditioning on $y_{0:t}$, we obtain the expression

$$w_t(x_{0:t} \mid y_{0:t}) = \frac{p(x_{0:t} \mid y_{0:t})}{g(x_{0:t} \mid y_{0:t})}$$

We then proceed in a similar fashion as when deriving (2.9). This time we consider $g(x_{0:t} \mid y_{0:t}) = g(x_t, x_{0:t-1} \mid y_{0:t})$, to obtain the factorization

$$g(x_{0:t} \mid y_{0:t}) = g(x_t \mid x_{0:t-1}, y_{0:t}) g(x_{0:t-1} \mid y_{0:t-1}).$$

Inserting (2.10) into (2.9) yields the following expression for the joint posterior distribution

$$\begin{aligned} p(x_{0:t} \mid y_{0:t}) &= \frac{p(y_t \mid x_t) q(x_{t+1} \mid x_t) p(x_{0:t} \mid y_{0:t})}{p(y_t \mid y_{0:t-1})} \\ &\propto p(y_t \mid x_t) q(x_t \mid x_{t-1}) p(x_{0:t-1} \mid y_{0:t-1}), \end{aligned}$$

which exhibits the same recursive dependency as our factorization of g. We have dropped the one-step likelihood $p(y_t \mid y_{0:t-1})$ from the expression, which is perfectly fine as we are using a self-normalized IS scheme. This means that the distribution is only known up to a normalizing constant.

Inserting the expressions for p and g into the definition of w_t we obtain the following *weight update formula*

$$w_t(x_{0:t} \mid y_{0:t}) \propto \frac{p(y_t \mid x_t)q(x_t \mid x_{t-1})}{g(x_t \mid x_{0:t-1}, y_{0:t})} w_{t-1}(x_{0:t-1} \mid y_{0:t-1}) \tag{2.11}$$

Through step 3. of the SIR algorithm, the weights associated with the generated sample are normalised. This step ensures equality in (2.11) and hence completes the derivation of the algorithm.

2.3.9 The bootstrap filter

Algorithm 2 can be used together with an initial distribution $\chi(x_0)$ to sequentially update the estimated joint posterior distribution $p(x_{0:t} \mid y_{0:t})$ as t increases.

A commonly used filtering scheme is the *bootstrap filter*. In this scheme, we assume that the proposal density is the same as the transition prior. That is, $g(x_t \mid x_{0:t-1}, y_{0:t}) = p(x_t \mid x_{t-1})$. This is a standard trick to simplify the computations and limit the number of assumptions needed. The primary drawback of using this approach is the additional variance this introduces in the estimator, due to frequent re-sampling being required (see below). An alternative approach would be to use the so-called *optimal proposal distribution*, which means using the target distribution as the proposal. However, sampling from that is often too computationally expensive for being practical. Using the transition prior as the proposal leads to the weight updating step being reduced to

$$\tilde{w}_t(x_{0:t} \mid y_{0:t}) = p(y_t \mid x_t) w_{t-1}(x_{0:t-1} \mid y_{0:t-1}),$$

where $\tilde{w}_t(x_{0:t} \mid y_{0:t})$ denotes the unnormalized weights.

However, even with self-normalization, the weights will rapidly drop to zero, making this unusable for anything but really small values of t. The reason for this is the built-in diffusion in the algorithm. The particles move freely around the state-space with no consideration taken to the usefulness of their current position. In statistics' terms, the variance of the estimator is growing *unboundedly*.

The bootstrap filter handles this problem by introducing a multinomial re-sampling step at each timestep. This re-sampling is carried through by drawing N particle trajectories from the density formed by the particle weights, i.e. according to their associated weights. In this way, we will get rid of the particles with low probability and only keep the relevant ones. As mentioned above, re-sampling comes at the cost of adding variance to the estimator. Introducing re-sampling and an initial draw of particles, we arrive at the following well-known algorithm

Algorithm 3: Bootstrap filter

Data: $y_{0:t}$
Result: ϕ_k^N for $k = 0, \ldots, t$
Draw $\{\xi_0^i\}_{i=1}^N \sim \chi$
Set $\{w_0^i\}_{i=1}^N = 1/N$
Define $\phi_0 = \{\xi_0^i, w_0^i\}_{i=1}^N$
for $k = 1, \ldots, t$ **do**
  Draw $\{\tilde{\xi}_{0:k-1}^i\}_{i=1}^N \sim \phi_{0:k-1|k-1}$
  Set $\{w_{k-1}^i\}_{i=1}^N = 1/N$
  Compute ϕ_k^N by feeding $\{\tilde{\xi}_{k-1}^i, w_{k-1}^i\}_{i=1}^N$ and y_k into Algorithm 2.
end

The output from Algorithm 3 can be used to formulate the *SMC estimator*. This estimator is defined by

$$\hat{\varphi}_t^N \stackrel{\text{def}}{=} \frac{1}{N} \sum_{i=1}^{N} w_t^i \xi_t^i$$

where $\{\xi_t^i, w_t^i\}_{i=1}^N$ is the weighted particle system. It can be shown that the SMC estimator is an MC estimator of x_t. A full theorem and proof for this can be found in [5].

2.3.10 Predicting the future

To find an analytically tractable closed-form expression for the posterior is, in general, not possible. However, in a true Monte Carlo spirit we can make use of our filtered marginals and substitute the analytical evaluation with an empirical distribution. Factoring $p(y_{t+1} \mid y_{0:t})$, we obtain the following expression

$$p(y_{t+1} \mid y_{1:t}) = \int p(y_{t+1} \mid x_{t+1}) q(x_{t+1} \mid x_t) \phi_t(dx_t) \tag{2.12}$$

To sample from $p(y_{t+1} \mid y_{1:t})$, we will use our filtered marginal distribution ϕ_t^N. Because of this trick, generating a sample of predicted values y_{t+1}^{pred} is simply a matter of drawing from each distribution one at a time, similar to Gibbs sampling. How to perform this sampling explicitly is defined in Algorithm 4. If the transition density and observation density are multivariate, further factorization might be required.

Because of the properties of the MC estimator discussed earlier, the resulting sample $\{y_{t+1}^{i,pred}\}_{i=1}^M$ can be considered a set of I.I.D. draws from the predictive distribution $p(y_{t+1} \mid y_{1:t})$.

2.3.11 Backward smoothing

In order to go back and solve the learning problem we will need a good approximation of the joint posterior distribution $p(x_{0:t} \mid y_{0:t})$. Unfortunately, we cannot use

Algorithm 4: Sampling from one-step predictor

Data: Filtered marginal distribution ϕ_t^N
Result: Sample of one-step predictions $\{y_{t+1}^{i,pred}\}_{i=1}^M$
for $i = 1, \ldots, M$ **do**
  Draw $\tilde{\xi}_t^i \sim \phi_t$
  Draw $\xi_{t+1}^{i,pred} \sim q(\xi_{t+1} \mid \tilde{\xi}_t^i)$
  Draw $y_{t+1}^{i,pred} \sim p(y_{t+1} \mid \xi_{t+1}^{i,pred})$
end

the sequence of filtered distributions ϕ_t^N from the bootstrap filter. Because of the re-sampling step, all particles will share the same trajectory up until only the final couple of time steps.

This phenomenon is called *path degeneracy* and causes large errors if the filtered particle trajectories are used as an approximation of the whole joint posterior distribution. To address this, we will need to perform something a so-called *backward pass*. At time T, using all of these individual filter marginals, we will attempt to recover the smoothed joint posterior distribute using a method called *backward sampling*. This method has been treated extensively in, e.g. [7], where convergence and other properties are also discussed. Starting with the last filtered

Algorithm 5: Backward sampling algorithm

Data: ϕ_t^N for $t = 0, \ldots, T$
Result: $\phi_{0:T|T}^M$
Draw $\{\tilde{\xi}_T^i\}_{i=1}^M \sim \phi_T^N$
Define $\phi_{T|T}^M$ as $\{\tilde{\xi}_T^i\}_{i=1}^M$
for $t = T-1, \ldots, 0$ **do**
  for $k = 1, \ldots, M$ **do**
    for $j = 1, \ldots, N$ **do**
      Compute $w_{t|t+1}^j = q(\tilde{\xi}_{t+1}^k \mid \xi_t^j) w_t^j$, where ξ_t^j, w_t^j from ϕ_t^N
    end
    Normalize weights, s.t. $\sum_{i=1}^N w_{t|t+1}^j = 1$
    Choose ancestor $\tilde{\xi}_t^k = \xi_t^j$ with probability $w_{t|t+1}^j$
  end
  Obtain $\phi_{t:T|T}^M$ by adding $\{\tilde{\xi}_t^i\}_{i=1}^M$ to $\phi_{t+1:T|T}^M$
end

marginal distribution ϕ_T^N, suitable ancestors $\{\tilde{\xi}_{T-1}^i\}_{i=1}^M$ are selected from the previous filtered marginal distribution ϕ_{T-1}^N. This is then repeated recursively for remaining times $t = T-1, \ldots, 0$ to obtain the joint smoothed posterior density $\phi_{0:T|T}^M = \{\tilde{\xi}_{0:T}^i, w_T^i\}_{i=1}^M$. We have provided the full algorithm in Algorithm 5.

2.3.12 Sequential Monte Carlo expectation-maximization

Once we have found a way to compute the joint smoothing posterior density $\phi_{0:T|T}$, we can start looking for a way to compute the optimal values for the hyperparameters θ. Going back to the EM algorithm, we will need make a few adaptations to get it to work under a particle-based regime.

In this thesis we will use the *Sequential Monte Carlo Expectation-Maximization* (SMCEM) algorithm discussed in [17]. This paper provides insight to the validity of the algorithm under the SMC paradigm, along with interesting convergence results. In short, the algorithm operates on a batch of observations $y_{0:T}$. It resembles the standard EM algorithm (see Algorithm 1) in that first the auxiliary quantity $\mathcal{Q}(\theta, \theta')$ is computed and then the hyperparameter θ' is updated by finding $\arg\max_{\theta\in\Theta} \mathcal{Q}(\theta, \theta')$. These two steps are then repeated until optimality has been reached. The primary difference is that we do not have access to the true joint posterior distribution $p(x_{0:t} \mid y_{0:t})$, but instead have to rely on the *smoothed* joint posterior distribution $\phi^M_{0:T|T}$ for computing $\mathcal{Q}(\theta, \theta')$. This can lead to a considerably increased level of complexity. However, as long as the complete data likelihood function $p_\theta(x_{0:t}, y_{0:t})$ belongs to the exponential family, the procedure becomes straightforward. We can simply compute the sufficient statistics $S_T(\theta')$ from $\phi^M_{0:T|T}$ (which is computed under θ'), to approximate $\mathcal{Q}(\theta, \theta')$ by an M-particle approximation defined as

Algorithm 6: SMC Expectation-Maximization

Data: $y_{0:T}$
Result: θ^*
Set initial guess $\theta' = \theta_0$
while *Stopping criterion not met* **do**
  for $t = 1, \ldots, T$ **do**
    Compute and store $\phi^N_t(\theta')$ using Algorithm 3
  end
  Compute $\phi^M_{0:T|T}(\theta')$ by inserting all $\phi^N_t(\theta')$ into Algorithm 5
  Compute sufficient statistics $S_T(\theta')$ from $\phi^M_{0:T|T}(\theta')$
  Set $\mathcal{Q}^N(\theta, \theta') = \eta(\theta) \cdot S_T(\theta') - A(\theta)$
  Update $\theta' = \arg\max_{\theta\in\Theta} \mathcal{Q}(\theta, \theta')$
end
Set $\theta^* = \theta'$

$$\mathcal{Q}^M(\theta, \theta') = \eta(\theta) \cdot S_T(\theta') - A(\theta), \tag{2.13}$$

where η and A are the *natural parameter* and *log-partition functions*, respectively. When the distribution is known, finding the optimal θ' is easy. The full SMCEM algorithm is defined in Algorithm 6.

We note that, as discussed in the paper referenced above, for the SMCEM algorithm to converge, it is required to increase the number of particles with each iteration. This is typically done at a quadratic increase rate. Further, to obtain good convergence results, several hundreds of iterations might be required.

Chapter 3

Model

In this chapter we define the *scaled volume imbalance* Ψ, which is an adaptation of the volume imbalance discussed in Section 2.1.2. We find that Ψ is an observable outcome of an underlying process. This process is carefully studied and modelled, resulting in an elegant hidden Markov model (see Definition 1).

3.1 The scaled volume imbalance

The volume imbalance, as discussed in Section 2.1.2, has many interesting properties. With the successful concave modelling of volume in relation to price impact, as discussed in Section 2.1.4, along with the positive effects this has on observed distributions, as we will discuss later in Section 3.2.3, we propose an adaptation of this quantity, which we will call the *scaled volume imbalance* Ψ. Moving forward, this is the quantity that we will study using the Monte Carlo framework that we develop in this thesis.

To formulate the definition of Ψ we will first define the *scaled volumes* ν via the concave transform

$$\nu = \sqrt{q} \tag{3.1}$$

of the volumes q associated with individual trades.

Without making any further assumptions at this stage, we say that Ψ is observed at time t by the observable outcome ψ_t, defined by

$$\psi_t \stackrel{\text{def}}{=} Q_t^B - Q_t^S = \sum_{i=1}^{n_t^B} \nu_{t,i}^B - \sum_{i=1}^{n_t^S} \nu_{t,i}^S \tag{3.2}$$

Here, t denotes the discrete timestep and index i denotes each (pooled) trade in the observed set of trades for that timestep, with n_t^B buyer-initiated and n_t^S seller-initiated trades, respectively. The values ν_i, are the scaled volumes, as defined in (3.1), associated with each trade.

The primary distinction between the scaled volume imbalance and the standard volume imbalance is the concave scaling of traded volumes.

Remark 1. The scaled volume imbalance only considers *executed trades*. This keeps down the complexity while at the same time a high signal-to-noise ratio is obtained. We will discuss this topic more in-depth in Section 6.6

Remark 2. There are, of course, many proposals for models that make use of all actions (see, for example, [11]). This generally leads to a very complex model with a multitude of unknown parameter, which is something that we want to stay away from here. Therefore, we have another reason to limit ourselves to looking at trades only.

3.2 Making assumptions

In this section we will discuss what assumptions can be made on the underlying generation process of executed trades and their attributes of interest. All assumptions are motivated in detail. Where possible, we provide empirical evidence to support our choices.

3.2.1 Trade generation is memoryless

If we consider the *limit order flow*, i.e. regular placing of orders, this is sometimes assumed to be memoryless in order to get analytically tractable solutions. However, thinking about it, it is easy to imagine that market participants act on limit orders that are hitting the order book and as a result place their own limit orders. There is actually a technique called *spoofing*, by which traders use deceptive orders to bait other traders into trading. This kind of scheme is illegal, but through its existence, the technique invalidates the memorylessness assumption for limit orders.

On the other hand, when it comes to *trades*, these actions do not introduce any new information, as discussed in [3]. Therefore, there is no reason for a market participant to act on an executed trade. In reality, for someone who wants to buy (who believes the price to be fair or too low) an incoming buy-initiated trade can only cause the trader *not to buy*—by, for example, taking all the liquidity on the best ask level. "Not not buy" is not an action and, hence, does not invalidate the memorylessness assumption.

Before concluding that the memorylessness property can be used to describe trade generation, there is, however, something else we will first need to consider. Most electronic trading platforms support splitting of a market order to match against multiple limit orders, if the full volume cannot be executed against the single limit order with highest priority. This will cause a single trade order to result in multiple simultaneous trade executions. Hence, observing each execution as if it were a unique trade would violate the memorylessness assumption.

Further, the high level of automation associated with high-frequency markets means that many traders are using computer algorithms, which are (typically) based

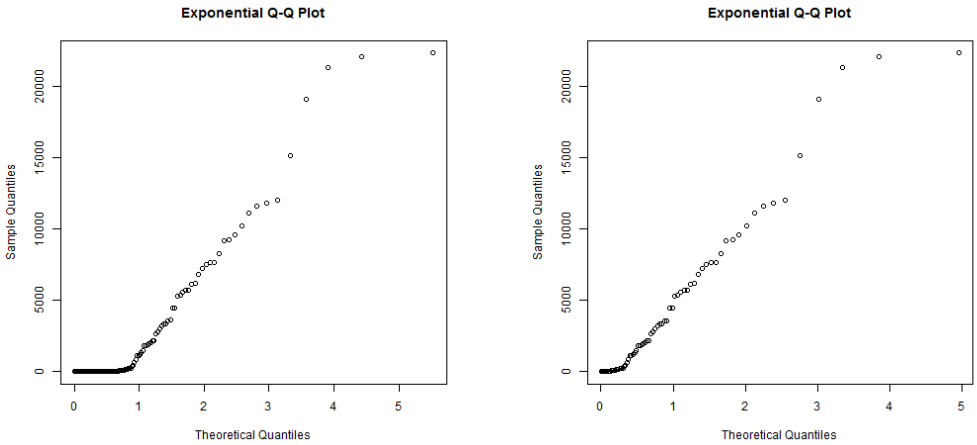


Figure 3.1. Inter-arrival times of sell trades during 8 minutes of trading, without pooling (left) and with pooling (right). Vertical axis displays values in milliseconds.

on a set of deterministic rules. It happens that different traders use very similar, if not identical, algorithms. This is mathematically equivalent to having one single trader—one belief system—performing several correlated trades virtually at the same time.

To address both of these problems, we propose the introduction of something we call *trade pooling*. This is a procedure in which trades that are temporally very close are pooled together, to count only as a single trade. The volume of a pooled trade is defined by the sum of the volumes of all trades that together make up the pooled trade—just as if the trade was only one larger trade rather than several smaller ones. Using this concept we make the following assumption

Assumption 1. *The generation process of pooled trades for a particular side (Buy or Sell) is memoryless.*

To motivate this, we need to realize that the issues pointed out above both result in simultaneous trades. In the first problem we will see trades with the exact same timestamp, whereas in the second problem there might of course also be some associated latencies. Therefore, trade pooling should definitely, at least, reduce these phenomena.

Further, studying this empirically we can see evidence that trade pooling really addresses this problem. After the trade pooling, the inter-arrival times actually display a strong exponential character, where before the pooling they did not. In Figure 3.1 we can see a comparison between exponential Q-Q plots for inter-arrival times between sell trades for 8 minutes of trading in the super-liquid front-month index futures contract FDAX Jun14 on April 28, 2014, with and without trade pooling. From this, for the rest of the thesis we will always refer by ”trade” to *pooled trades*.

3.2.2 Trade generation is time-dependent

We will proceed by making the following assumption

Assumption 2. *The generation process of pooled trades for a particular side (Buy or Sell) is time-dependent.*

This assumption can easily be motivated by intra-day seasonality effects that are readily observable, such as diurnality or that everybody takes a lunch break at the same time, causing a sudden decrease in trading activity. The trading intensities are also highly sensitive to news releases and other information generators.

3.2.3 Scaled volumes are exponentially distributed

The distribution associated with volumes of individual trades has been studied a lot and is generally assumed to follow a power-law (see, for example, [13]). Drawing from this knowledge, the transformation defined in (3.1), which is the inverse to the described power-law, takes away the problematic fat tails of the distribution and hence reveals more of the inner features. We formulate the first of those features by the following assumption.

Assumption 3. *The scaled volumes ν for pooled trades on a particular side (Buy or Sell) are exponentially distributed.*

We will try to motivate this assumption by providing some empirical evidence. In Figure 3.2 we have applied the transform to the volumes of pooled trades executed during 30 minutes of trading in the super-liquid front-month futures contract *FDAX Jun14* on April 28, 2014. As we can see the scaled volumes exhibit a pronounced exponential behaviour.

It should be noted that it might seem a bit counter-intuitive to use a continuous distribution such as the exponential distribution to describe something as clearly discrete as the the scaled volumes, instead of using a discrete distribution. However, due to the non-linearities in the outcomes it would be very problematic to find a suitable discrete distribution that characterizes this behaviour. Also, as we will see later, in this thesis we will only consider sums of scaled volumes, which approach the continuous case.

3.2.4 The scaled volume distribution is time-dependent

Similar to the distribution of trade generation, we formulate the following assumption

Assumption 4. *The distribution of scaled volumes ν for pooled trades on a particular side (Buy or Sell) is time-dependent.*

This assumption is primarily introduced for symmetry. It can be motivated, for example, by the possibility of an institutional trader entering the market at a certain point of the day, raising the average volume sizes.

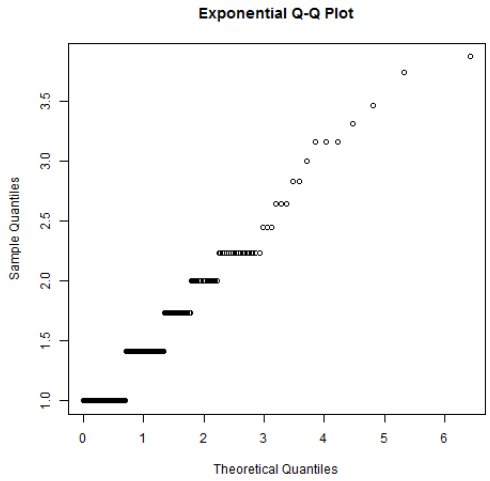


Figure 3.2. Exponential Q-Q plot for scaled volumes of pooled sell trades in FDAX Jun14 during 30 minutes of trading on April 28, 2014

Further, when repeating the study in the previous section for other times of the day, the parameter of the exponential distribution varies when fitted to data for the different times, which is an empirical indicator to motivate this assumption.

3.3 Defining the model

Now that all the assumptions and the theoretical parts are laid out, we are ready to define the model to use for making inference about Ψ. Trying to capture the intra-day seasonality, as well as allowing for news events and similar, we will describe Ψ in terms of an HMM, as defined in Definition 1. We will see that such a state-space model appropriately reflects the nature of the financial markets. The observations will be things like number of trades and the traded volumes, while the latent variables, the *state*, will be associated with hidden processes driving the dynamics of the markets.

3.3.1 The observations

To obtain the observation density we must start by defining the set of observations that we will be using. We could limit ourselves to only consider the observed values ψ_t directly. However, doing so would cause a lot of the available information to be lost. Diving into the components that make up Ψ, we realize that we have access to the outcomes of the scaled aggregations Q^B and Q^S, too. Examining the Q processes themselves closer, we have the following definition of their observable outcomes

$$Q_t^{(*)} = \sum_{i=1}^{n_t^{(*)}} \nu_{t,i}^{(*)},$$

where $(*)$ denotes the *side* (Buy or Sell) and t denotes a particular time step.

Combining Assumption 1 with Assumption 2, we realize that the value $n_t^{(*)}$ can be modelled as an outcome of an inhomogeneous Poisson process, as defined in Section 2.2.3. Hence, $n_t^{(*)}$ is a Poisson distributed outcome for some Poisson parameter $\lambda_t^{(*)}$ on the time interval $[t, t + \Delta t)$.

Further, by Assumption 3 and Assumption 4, the scaled volumes ν_i can each be modelled as an outcome of an exponential distribution with some time-dependent scale parameter μ_t.

Since it is possible to observe both $n_t^{(*)}$ and the associated scaled volumes $\{\nu_{t,1}, \ldots, \nu_{t,n_t^{(*)}}\}$, we will try to formulate a model that makes use of all the available information—not only ψ_t. To accomplish this, we define the observation y_t as

$$y_t = \{n_t^B, n_t^S, Q_t^B, Q_t^S\} \tag{3.3}$$

The reason that we choose to not observe each individual scaled volume, is that due to the memorylessness property of the exponential distribution, no information is provided by the individual outcomes in addition the that of their total sum.

The sum of a known number exponentially distributed random variables having the same parameter, is called the *Erlang distribution*, which simply is a Gamma distribution with an integer-valued shape parameter. This leads to the following relations

$$\begin{aligned} n_t^B &\sim \text{Po}(\lambda_t^B) \\ n_t^S &\sim \text{Po}(\lambda_t^S) \\ Q_t^B \mid n_t^B &\sim \text{Erlang}(n_t^B, \mu_t^B) \\ Q_t^S \mid n_t^S &\sim \text{Erlang}(n_t^S, \mu_t^S) \end{aligned} \tag{3.4}$$

These relations together make up the observation density $p_\theta(y_t \mid x_t)$. Since the outcomes are independent (apart from the conditioning on $n_t^{(*)}$ in $Q_t^{(*)}$) the full observation density can be written as

$$p_\theta(y_t \mid x_t) = f(Q_t^B \mid n_t^B, \mu_t^B) f(n_t^B \mid \lambda_t^B) f(Q_t^S \mid n_t^S, \mu_t^S) f(n_t^S \mid \lambda_t^S) \tag{3.5}$$

where f represents the probability density functions associated with each of the distributions in (3.4).

As we can see there is no dependency on any external hyperparameters θ, which means that we have $p_\theta(y_t \mid x_t) = p(y_t \mid x_t)$, and hence we drop θ from the notation.

3.3.2 The latent variables

From the definition of the observations y_t (see Equation 3.3) the set of latent variables x_t to use in this HMM comes out naturally as

$$x_t = \{\lambda_t^B, \lambda_t^S, \mu_t^B, \mu_t^S\} \tag{3.6}$$

The trading intensities $\lambda_t^{(*)}$, for buyer-initiated and seller-initiated trades, are time-dependent as per Assumption 2. They are even expected to fluctuate quite a lot throughout the day, as noted in the discussion in connection to that assumption. To cope with these fluctuations, we will model the transitions by a Laplace distribution. The Laplace distribution has some nice features; it is leptokurtic (to allow for sudden jumps), belongs to the exponential family and only requires a single parameter b when assumed to be centered around zero. As we have no reason to assume drift in either direction for the trading intensities, this distribution fits our purpose very well.

By Assumption 4 we also assume that the scale parameters associated the volume distributions $\mu_t^{(*)}$ are time-dependent. However, in this case, we do not expect them to change very much. We will therefore model these changes by a good old Normal distribution. Following the same reasoning as for the intensities $\lambda_t^{(*)}$, we assume no drift and hence only have one parameter to fit, the standard deviation σ. This can be summarized by

$$\begin{aligned}
\lambda_{t+1}^B &\sim \text{Laplace}(\lambda_t^B, b^B) \\
\lambda_{t+1}^S &\sim \text{Laplace}(\lambda_t^S, b^S) \\
\mu_{t+1}^B &\sim \mathcal{N}(\mu_t^B, \sigma^B) \\
\mu_{t+1}^S &\sim \mathcal{N}(\mu_t^S, \sigma^S)
\end{aligned} \tag{3.7}$$

Further, in this thesis, we have assumed that the transitions are all independent between the latent variables, i.e. $\lambda_{t+1}^B - \lambda_t^B \perp \lambda_{t+1}^S - \lambda_t^S \perp \mu_{t+1}^B - \mu_t^B \perp \mu_{t+1}^S - \mu_t^S$.

In contrast to our observation density, the transition density depends on a set of hyperparameters $\theta = \{b^B, b^S, \sigma^B, \sigma^S\}$. Since the transition kernels are symmetric, θ is purely describing the rate of the evolutionary diffusion.

3.3.3 The initial distribution

We will not discuss the functional form of the initial distribution $\chi(x_0)$ as it is very difficult to formulate any useful assumptions on distribution characteristics from a market microstructure point-of-view. Fortunately, the initial distribution is not very interesting as the result will (asymptotically) not depend on it.

In such cases, it is generally advised to look for more general initial distributions that come with good properties for making statistical inference. However, we have found that for the purpose of this thesis it is sufficient to simply instantiate all particles to some constants, $x_0 = \{a, b, c, d\}$, and then mutate them one time according to q_θ.

Chapter 4

Method

In this section we will discuss how the hidden Markov model defined in Chapter 3 can be studied using a particle filter based approach as described in Section 2.3 of Chapter 2.

4.1 Framework

In this section we will propose a Monte Carlo framework for solving the *learning problem* and *inference problem* related to an HMM describing market microstructure phenomena in a high-frequency setting. The quantity of interest in our study is the scaled volume imbalance Ψ as defined in Chapter 3.

4.1.1 On-line inference

A bootstrap filter (Algorithm 3) is implemented to track the latent variables $x_t = \{\lambda_t^B, \lambda_t^S, \mu_t^B, \mu_t^S\}$, using the observations $y_t = \{n_t^B, n_t^S, Q_t^B, Q_t^S\}$. With the objective in this thesis focusing on making (one-step) predictions and performing model learning through the SMCEM algorithm, we will only need to consider the sequence of filtered marginal distributions ϕ_t^N for $t = 0, \ldots, T$. The observations are sampled on consecutive sampling periods of length Δt, in which executed trades are *pooled* if they are executed within a very small time-span τ of each other, as discussed in Section 3.2.1.

The transition kernel $q_\theta(x_t \mid x_{t-1})$ used for the particles is taken to be the one with Laplacian (for $\lambda_t^{(*)}$) and Gaussian (for $\mu_t^{(*)}$) densities, as discussed in Section 3.3.2.

The likelihood function used to update the weights for each observation is defined by the observation density in (3.5).

The proposal distribution in the SIR algorithm (Algorithm 2) is taken to be the same as the transition density, i.e. $g(x_t^i \mid x_{t-1}^i, y_t) = q(x_t^i \mid x_{t-1}^i)$, as per the definition of the bootstrap filter.

4.1.2 Model learning

Learning is done by an implementation of the powerful SMCEM algorithm (Algorithm 6). We consider each trading day a separate batch of observations. This is a natural way for splitting the data because we do not have to consider the potential jumps that can happen between closing and opening. Hence, we always run the SMCEM end-of-day using all the observations for that day. For each iteration we begin with a forward pass using the bootstrap filter to compute all marginals ϕ_t^N for $t = 0, \ldots, T$, then we proceed with a backward pass using the backward sampling algorithm (Algorithm 5) to simulate the joint smoothed posterior density $\phi_{0:T|T}^M$ from the marginals. The joint smoothed posterior is then used for completing the current SCMEM iteration.

In our case the only dependency on θ lies in the transition kernel q_θ. Since the dynamics of the latent variables are independent and we only assume diffusion, maximizing the auxiliary function $\mathcal{Q}$ only comes down to computing the complete data maximum likelihood estimators for each individual parameter in θ. In other words, we find the $\{b^B, b^S, \sigma^B, \sigma^S\}$ which best describes the smoothed particle trajectories in $\phi_{0:T|T}^M$, from a likelihood perspective.

After updating θ', we proceed with the next iteration until the MLE of θ is found—or at least a good estimate $\hat{\theta}$ of it. For the first 10 iteration we use $N = 1000$ and $M = 100$. After that, we increase the number of particles for each iteration quadratically. In this thesis we have not studied optimal stopping conditions. Instead we simply consider θ optimal after 20 full iterations. At this point we have $N = 2000$ and $M = 200$. We note that the appropriateness of this stopping criterion is supported by Figure 5.1.

Based on the scheme defined above, we will use the hyperparameter estimated from the observations on day $k - 1$ (yesterday) for making on-line predictions on day k (today). We denote the approximation of today's optimal hyperparameter θ_k^*. Thus, with this notation, we have that $\theta_k^* = \hat{\theta}_{k-1}$.

4.1.3 Model assessment

The quantity of interest in this thesis is the scaled volume imbalance Ψ. The set of observables $y_t = \{n_t^B, n_t^S, Q_t^B, Q_t^S\}$ is merely a construction emerging from the proposed model for tracking Ψ. Therefore, for all model assessment we will restrict the performance analysis to checks that are addressing ψ_t directly. This is the same as setting the function T in (2.6) to be ψ_t from (3.2), i.e. $T : y_t \mapsto \psi_t$, or

$$T(y_t) = Q_t^B - Q_t^S. \tag{4.1}$$

With a clear view on what we want to study, we begin the model assessment with the second item in Section 2.2.9. For this we will look at the posterior distribution for some timesteps to see that it behaves sensibly and is not degenerated or otherwise pathological. After this sanity check, we will move on to examining the prior as

per the first item. This will be done by looking at the performance of the inference when using non-optimal values of θ.

For the third item, *fitness to data*, prediction is carried out via Algorithm 4. In short, we draw a sample of predicted observations y_{t+1}^{pred} from the predictive distribution $\hat{\phi}_{t+1|t}^{N}$. This predictor is obtained by simply mutating the filtered marginal ϕ_t^N one time. We will repeat this across the day, obtaining a sequence of predictive distributions $\hat{\phi}_{t+1|t}^{N}$ for $t = 0, \ldots, T$.

Using the predicted observations we will analyse the probability that the predicted scaled volume imbalance Ψ_t^{pred} take a value less than or equal to the observed outcome ψ_t. By using the expression for T above, this is the same as computing the posterior p-values, as defined in Equation 2.6, for each time step t. This gives us the sample $\{\mathbb{P}(\Psi_t^{pred} \leq \psi_t \mid y_{0:t-1})\}_{t=1}^{T}$ that we denote by $\{u_t\}_{t=1}^{T}$. Here, we realize that by assuming that the predictive distribution actually is the *real distribution* of Ψ_{t+1}, we can consider $\{u_t\}_{t=1}^{T}$ to be the cumulative probabilities for the observed outcomes ψ_{t+1}.

It should be noted, though, that this assumption is only valid if the transitions in x between two time-steps are relatively small, since the dynamics directly affects the posterior predictor by adding variance. Seeing that this is sensible, we use this assumption to check the performance of our predictions. If the assumption holds, the computed p-values for each timestep should together be I.I.D. draws from a uniform distribution, by the *Probability Integral Transform*.

The primary check for uniformity in the p-values will be a Uniform Q-Q plot of $\{u_t\}_{t=1}^{T}$ for each day. Another property that we will need to check is that there are no clustering effects in the predictions, i.e. that we do not see longer sequences of what is considered "improbable outcomes". For this, we will also look at the distribution of the differences $\Delta u_t = u_t - u_{t-1}$ against the theoretical distribution of a difference of two Uniform random variables. That distribution has the density function $p(\Delta u_t) = (1 - |\Delta u_t|)/2$.

We will also show how to perform a formal hypothesis testing of the predictor. Looking at the $1 - p$ confidence interval of the predictor we define the following *exceedance indicator*

$$I_t = \begin{cases} 0, & \text{if } \Psi_t^{pred;p/2} \leq \psi_t \leq \Psi_t^{pred;1-p/2} \\ 1, & \text{otherwise} \end{cases} \tag{4.2}$$

where $\Psi_t^{pred;q}$ indicates the quantile function of the posterior predicted distribution and p is typically set to 5%. For a batch of T observations, the sum $\sum I_t$ should then follow the Binomial distribution for T trials with probability p. From this relation, we can conduct two types of tests. To test that the predictor neither underestimates, nor overestimates the dynamics, we will use a *two-sided test*.

Using a *single-sided test*, this device can be used for *anomaly detection* of trading days with unpredictably high dynamics.

4.2 Implementation

In this section we will provide some tips and tricks and discuss some of the specific choices we have made for the implementation of the algorithms used in this thesis.

4.2.1 Implementation Details

All algorithms have been implemented in Java 8. To use as pseudo-random number generator (PRNG) the WELL19937c implementation from Apache Commons Math 3 was chosen. This PRNG is well suited for Monte Carlo methods because it has an extremely long period, compared to e.g. the standard JDK generator. Without regard to the choice of programming language, it is always advised to make sure an adequate PRNG is being used.

Draws from distributions having a closed-form inverse are performed using the Inverse Transformation Theorem. For other distributions, such as the Poisson distribution and the Gamma distribution, Apache Commons Math 3 is used.

The bootstrap filter is run single-threaded since it is, in essence, sequential; the normalization step requires all particles to be known. It would be possible to make some parts of it multi-threaded, but that was not deemed necessary. However, running backward sampling single-threaded was a bit slow. There are other backwards smoothing schemes, aiming to reduce the complexity, but since implementing multi-threading for this algorithm was straightforward, we decided that this was sufficient. To make it multi-threaded with n threads, at each timestep the children are split into n equal sized batches, handled each by their own thread. This made the computation speeds fall into good time-scales.

4.2.2 Specifics

When implementing a framework of this scale there are a lot of small choices that must be made.

For the the SMCEM algorithm we decided that it was sufficient to formulate our stopping criterion as "perform 20 iterations in total". The algorithm converges very rapidly, as can be seen in the Figure 5.1, so these 20 iterations will suffice. The error from using the previous day's estimated hyperparameters instead of today's values will be much more significant. Further, as an initial guess for the θ, we have used $\theta' = \{5.1, 7.4, 0.46, 0.33\}$ across all days and instruments successfully.

We have not aimed to capture the dynamics of off-hour trading. In some markets, trades can be reported after trading has closed. These trades are usually OTC trades, or in some other way not directly related to the electronic trading. Therefore, we have restricted ourselves to only consider trades within the regular trading day. We have also decided to not consider auction trades.

When running the SMCEM algorithm for learning the hyperparameters, for the first 10 iterations 1000 forward particles and 100 backward particles were used. After that, the number of particles were increased quadratically, in the way that

at iteration 200 there were 2000 forward particles and 200 backward particles. For prediction, 1000 forward particles were used across all data runs.

4.2.3 Performance

All computations have been run on a PC with an Intel Core i5-6600K processor, 16GB RAM and Windows 10 64-bit operating system.

The algorithms used in this thesis are all linear in T. For $\Delta t = 1$ min, i.e. about 540 timesteps in a regular trading day, a forward pass with the bootstrap filter takes about 15 seconds and the backward pass with the backward sampling algorithm about 8 seconds. This means that a full iteration of the SMCEM algorithm takes about 23 seconds to complete, since the computational time needed for the M-step is negligible.

Consequentially, running for half the sampling interval ($\Delta t = 30$ s) takes twice the time; around 45 seconds. Similarly, setting $\Delta t = 10$ min results in iterations taking only 2–3 seconds to complete. Altogether, this makes the algorithm very potent and allows for a fair bit of extra complexity to be added in the future.

Chapter 5

Results

In this thesis we are studying data that Deutsche Börse AG have let us use for the purpose of this thesis. Deutsche Börse was early to embrace the digital era and is therefore now one of the leading actors in the world when it comes to electronic trading. Their main platforms are all electronic, using cutting-edge technology for managing their data, resulting in the finest quality data.

We have sampled the observations using the *Calculation Server* module for the software *Scila Surveillance* that Deutsche Börse uses for monitoring all trading activity in their platforms. We have focused on equities and futures instruments as they are the best understood contracts from a LOB perspective, making the results easy to generalize.

The specific instruments that will be studied are stocks for Adidas AG and Nordex AG, which are traded on the platform *Xetra*, together with the super-liquid front-month futures contract FGBM Mar16 ('Euro-Bobl Futures'), which is traded on the platform *Eurex*. The instruments are studied for days in the period from Feb 1–12, 2016.

5.1 Learning the hyperparameter

To show that the framework we have developed is capable of making inference, we must start by making sure that we are making adequate learning of the hyperparameter θ.

5.1.1 Convergence of the SMCEM algorithm

We start by verifying that the SMCEM algorithm produces consistent results. In Figure 5.1 we see the output of the SMCEM algorithm for trading on Feb 2 in Adidas AG and FGBM Mar16, respectively. For Adidas AG, the initial guess for θ was drawn uniformly between 1 and 8 for $b^{(*)}$ and between 0.05 and 1.2 for $\sigma^{(*)}$. For FGBM Mar16, the initial guess was drawn uniformly between 1 and 12 for $b^{(*)}$ and between 0.05 and 1.2 for $\sigma^{(*)}$. The algorithm was run 40 times, producing the

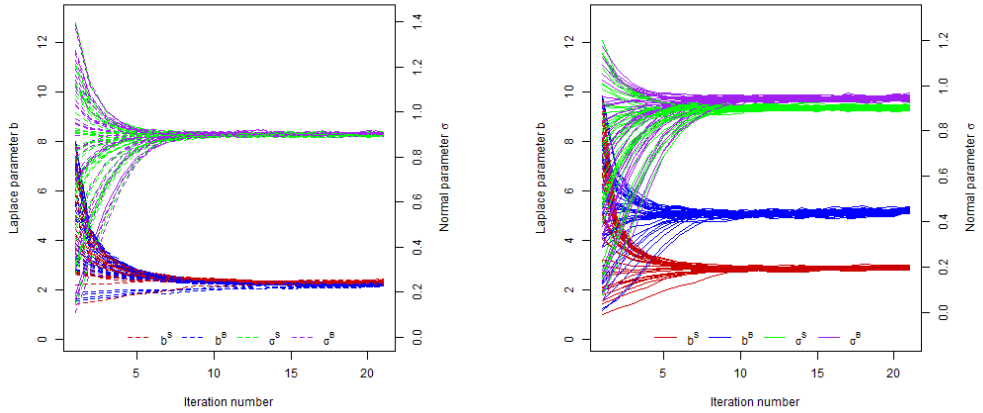


Figure 5.1. Estimates of optimal θ per iteration in the SCMEM algorithm when applied to trading on Feb 2, 2016, in stock Adidas AG (left) and futures contract FGBM Mar16 (right). The algorithm was run 40 times, each with 20 iterations.

40 trajectories presented in the figure. It is clear that the results are consistent. Further, we observe that the estimates approach the "end result" quicker from above than from below. This is expected, since it is easier to find a quiescent path among noisy trajectories, than to produce a noisy path from limited dynamics.

This figure shows that it takes about 10 iterations to get a decent estimate, even if the initial guess is quite a bit off. After 20 iteration, the estimate has stabilized further. For reference, in the FGBM Mar16 case the standard deviation for each component in θ is at this point only around 1% of its mean. For the purpose of this thesis this is considered sufficient. This motivates our choice of stopping criterion (see Section 4.2.2).

5.1.2 Hyperparameter dependency on Δt

Even though the quest for finding a perfect value of Δt goes beyond the scope of this thesis, we will need to verify that we do not encounter any problems because of a bad choice of sample interval length.

In Figure 5.2 we can see that the optimal values of the hyperparameters b^S increase with Δt. This behaviour has been found to apply in general for the b hyperparameters across the data. A possible interpretation is that the underlying processes scale accordingly for small Δt, but as Δt grows larger we see some kind of averaging affect. In other words, on a bigger time scale the underlying dynamics appear more stable.

Conversely, the optimal σ^S initially decreases with respect to Δt, to later become almost constant, as can be seen in Figure 5.3. The initial decrease can be be thought of as stabilization in regards to the observations. On the very small scale the average traded volume can change rather dramatically from one observation to the next since

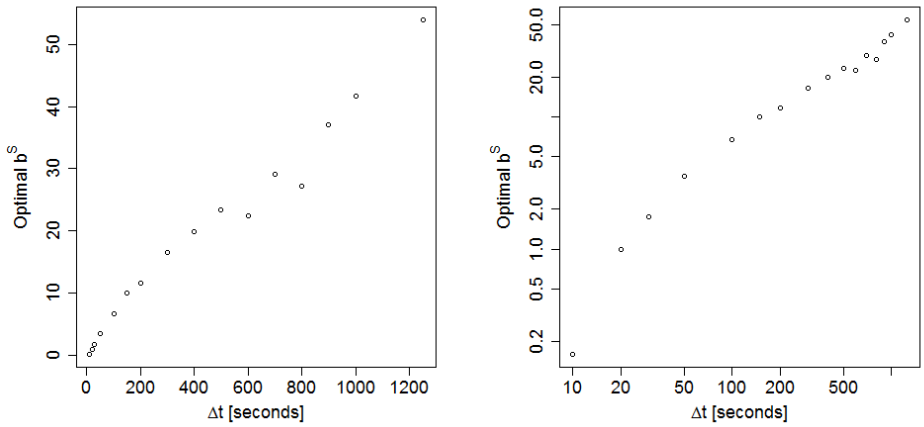


Figure 5.2. Standard plot (right) and log-log plot (left) for optimal value of b^S on Feb 4, 2016, for FGBM Mar16, against the length of the sampling interval Δt. The optimal b^S increases linearly at first, to then grow more slowly.

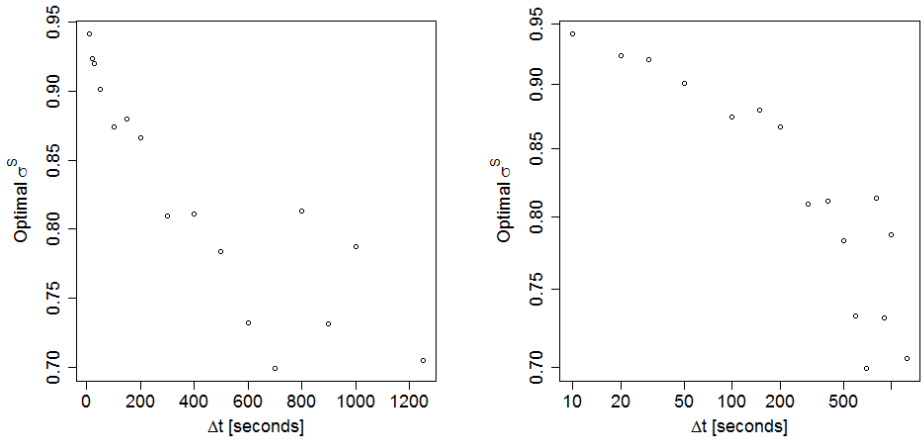


Figure 5.3. Standard plot (right) and log-log plot (left) for optimal value of σ^S on Feb 4, 2016, for FGBM Mar16, against the length of the sampling interval Δt. The optimal σ^S decreases at first, to then end up somewhere between 0.7 and 0.8.

each observation will only contain a few trades. This could lead to a little higher movement in μ than desired. As the sample length increases we therefore see a lower value for σ, leading to much smaller movement in μ_t.

In this thesis we have studied the results for both a smaller sampling interval of about 1 minute, and a longer sampling interval of about 10 minutes time. Looking the figures above, we believe that these choices for Δt should properly reflect the bigger picture.

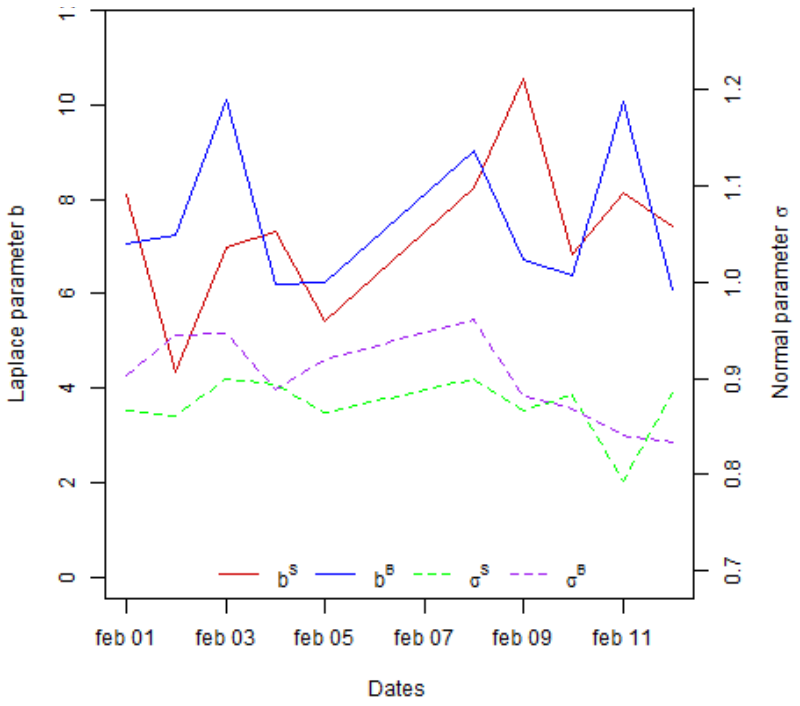


Figure 5.4. The optimal value of θ across multiple days for FGBM Mar16 using sampling time $\Delta t \approx 90$ s. Laplace parameters b are rather volatile while Normal parameters σ are fairly stable on a day-to-day basis.

5.1.3 Optimal values across multiple days

In Figure 5.4 the optimal values for the different components of θ are plotted over the period from Feb 1–12, 2016, for the front-month futures contract FBGM Mar16.

The observations were unfortunately sampled in such a way that Δt varies between 88 seconds and 93 seconds over the 2 week period. However, this does not affect the estimates significantly. Further, this result is primarily provided as a stepping stone for future discussions around how to improve the choice of θ_k^* discussed in Section 4.1.2.

In this figure we can see that the hyperparameters σ determining the rate of diffusion in μ_t are very stable. The change between days is less than 10%. On the other hand, the hyperparameters b determining the rate of diffusion in λ_t change the more. They seem to oscillate around some mean value but the process is quite volatile; the value is almost halved or doubled from one day to the next.

5.2 Parameter inference

The nature of the two types of latent variables, λ and μ, are radically different, which is clearly reflected in the results. In Figure 5.5 we are displaying typical

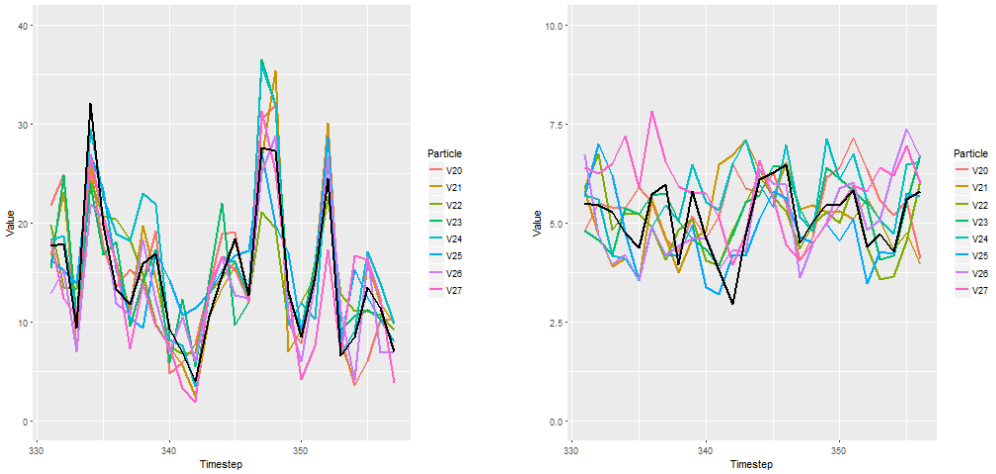


Figure 5.5. Smoothed particle trajectories of inferred λ_t^B (left) and μ_t^B (right) for 40 minutes of trading in FGBM Mar16. The black line represents the expected values. Observations were sampled at an interval $\Delta t = 90$ s. The λ_t^B values have small variance at each timestep, but are volatile with respect to t. The μ_t^B values have a lot higher variance for each timestep, but the mean is relatively stable, except for a quick drop during one step.

trajectories observed in the results. The trajectories in the plots are the smoothed trajectories for the buy parameters λ_t^B and μ_t^B during about 40 minutes of trading in the futures contract FGBM Mar16 for the second half of Feb 8, 2016. These results were obtained using a sampling time of $\Delta t = 90$ s.

As we can see in the left plot the smoothed marginals have quite low variance, while the expected values change from step to step. This is in line with our expectations, since the markets are known to fluctuate and can move fast at times (which we aim to capture with this model).

In the right plot we can observe the opposite characteristics being exhibited by the inferred μ_t^b. The particle trajectories are all over the place but the mean remains fairly stable, except for certain deviations on individual timesteps. It is likely that this variance is introduced as a side-effect of the rapid changes in λ_t when the smoothing is applied. Perhaps the significance of the time-dependency in μ_t in this model can be toned down, e.g. by penalizing μ_t dynamics or removing it completely, in order to improve the overall performance. As discussed in Section 3.3.2, the time-dependency was introduced to *allow* for intra-day changes in μ, not that it is necessarily expected.

5.3 Posterior predictive checks

The most important test is to verify that the N-particle predictive distribution $\hat{\phi}_{t+1|t}^N$ accurately describes the observed outcomes of Ψ_{t+1}. To show this we will look at the probability that the predicted outcome is lower than, or equal to the

observed outcome, $u_t = \mathbb{P}(\Psi_t^{pred} \leq \psi_t \mid y_{1:t-1})$, for each timestep t on a specific trading day. We are doing this for trading in three different financial instruments on Feb 3, 2016, each at two different sampling lengths Δt, the first being around 1 minute and the second around 10 minutes.

5.3.1 The predictive distribution

Even though we do not have a closed-form analytical expression for the posterior of Ψ, and even less for the posterior predictive distribution, by using the Monte Carlo machinery we can assess these distributions through their empirical distributions.

In Figure 5.6 we show the estimated predictive distribution step-by-step for a few predictions made in the futures contract FGBM Mar16 on Feb 8. For the purpose of this figure we have used the same-day MLE of θ to get a clear view of the behaviour of the predictor. Using any other approximation would affect the variance, since the dynamics are defined as pure diffusion.

We see that the predictor is looking quite symmetric and well-behaving; almost like a Normal distribution most of the time. A closer inspection of $t = 90$ and $t = 91$ further reveals the resemblance. For these particular time-steps, we have provided Normal Q-Q plots in Figure 5.7. This could be a side-effect of the mutation step when creating the predictive distribution.

Going through the histograms in Figure 5.6 one at a time we observe the following: In the first three timesteps the realized values are getting lower and lower. Before $t = 88$ there was a regime with high realized values of Ψ_t. At $t = 91$ the series of realizations have stabilized at a new normal level around 0, and so has the predictor, too, as can be seen in timesteps $t = 92$ and $t = 93$. Throughout the whole sequence of a falling Ψ the predictor maintained good precision on the predictions.

5.3.2 Uniformity in cumulative probabilities

Now we will focus on the performance of our "true" on-line predictor, i.e. using the on-line prediction algorithm with $\theta_k^* = \hat{\theta}_{k-1}$. This approximation will be used throughout the rest of the data analysis. Looking at Figure 5.8 we see that u_t are convincingly uniform in the distribution across all timesteps for each of the three stocks. There are tendencies showing that the predictions gets a bit more unreliable as Δt increases. This is perfectly intuitive as it simply means that predictions the longer into the future are a little harder to make.

For the FGBM Mar16 contract the predictor seem to have a bit higher variance than the actual distribution, resulting in higher number of observations than expected being closer to the center ($u = 0.5$) of the predictive CDF. The additional variance is introduced by the diffusion in the prediction step, and the excess in this case could be interpreted as that the FGBM Mar16 contract is more volatile in its dynamics on the 1 minute time-scale. Going back to the reasoning about dependency on Δt we realize that 1 minute is actually a longer horizon from an event

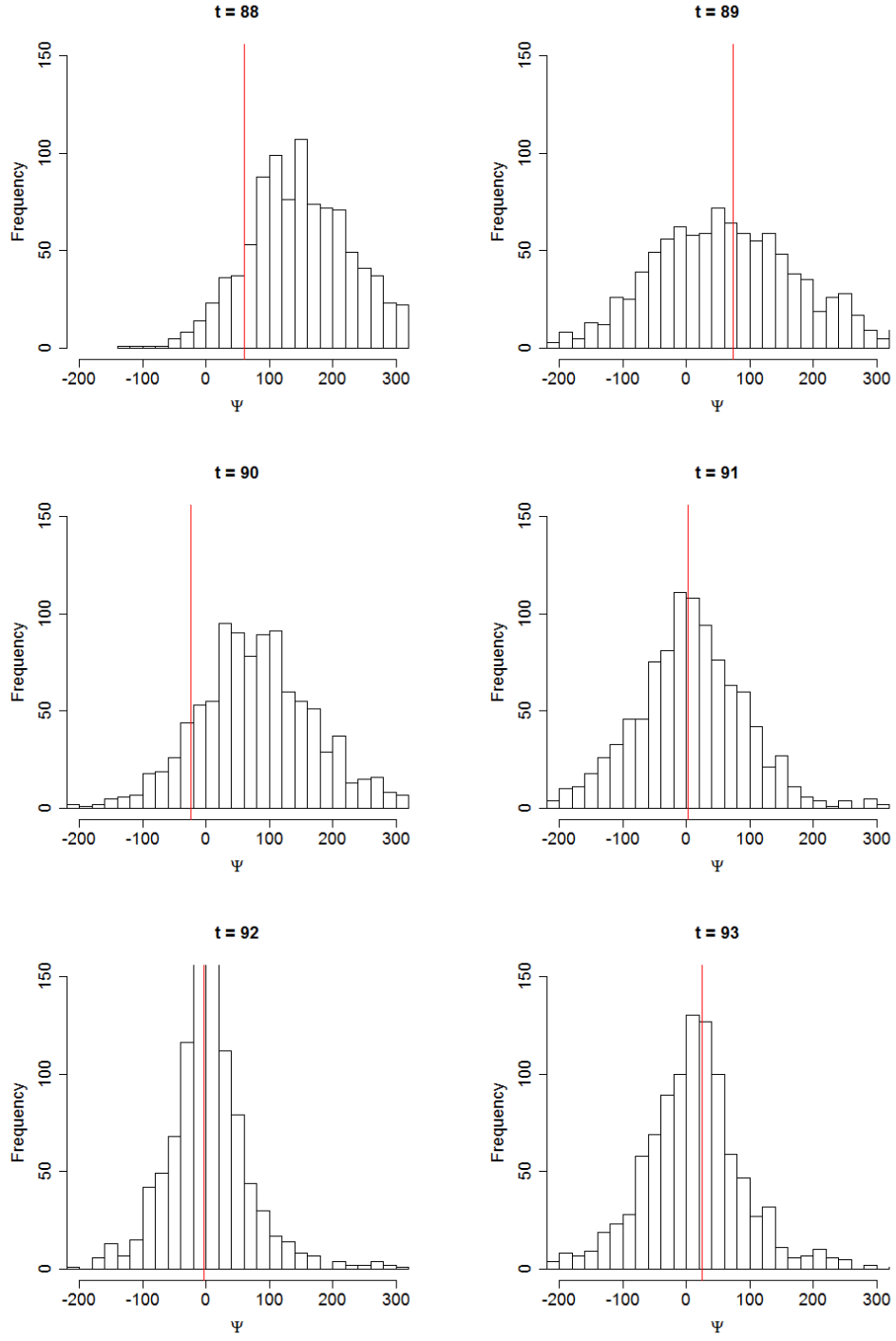


Figure 5.6. Histograms of predicted Ψ_{t+1} along with the actual observations ψ_{t+1} (indicated by the red lines) for 6 observations in FGBM Mar16 on Feb 8, 2017. The realized values are of high probability even in the event of rapid changes in the underlying regime.

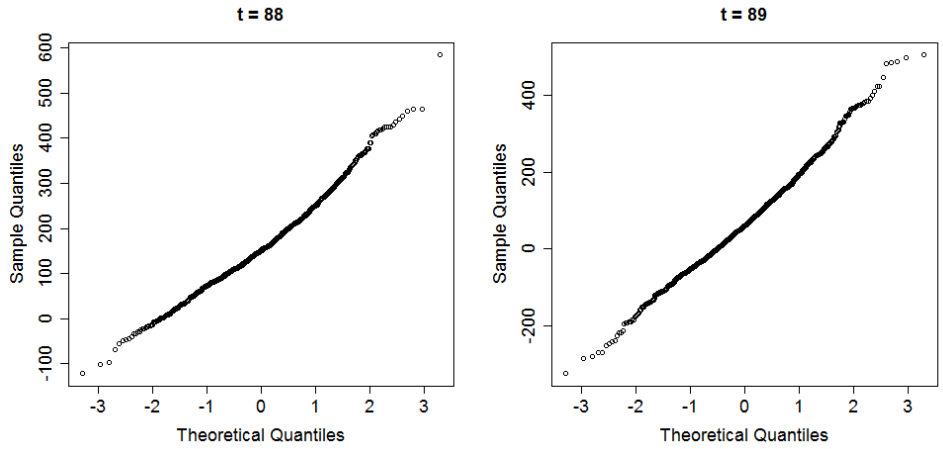


Figure 5.7. Normal Q-Q plots against the empirical distribution of the predicted Ψ_{t+1} for the timesteps $t = 88$ and $t = 89$ of trading in FGBM Mar16 on Feb 8, 2016. The predictive distributions have close resemblance to Normal distributions.

perspective for FGBM Mar16 than for the equities contracts because of its much higher liquidity.

In addition to the Q-Q plots, we will also perform two-sided hypothesis testing in relation to these predictions (see Section 4.1.3). We have only considered predictions on the shorter sampling interval and the results are summarized in the table below.

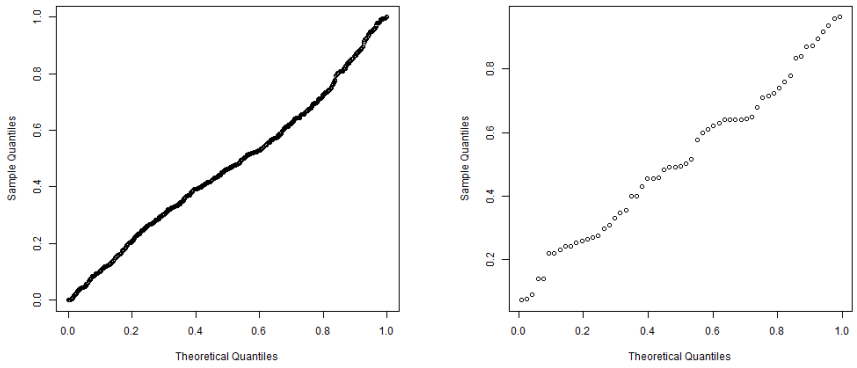
	Exceeds	Trials	p-value
FGBM Mar16	35	565	0.21
Adidas AG	14	458	0.054
Nordex AG	22	322	0.16

The number of observations, or *trials*, differ between the financial instruments for two reasons. The first being that the trading day for FGBM Mar16 is longer than for the stocks, and the second being that because of how the input data was structured, we had to use 90 second intervals for Nordex AG. We can see that in neither of the cases the null hypothesis can be rejected—indicating that our predictions are describing the observations well.

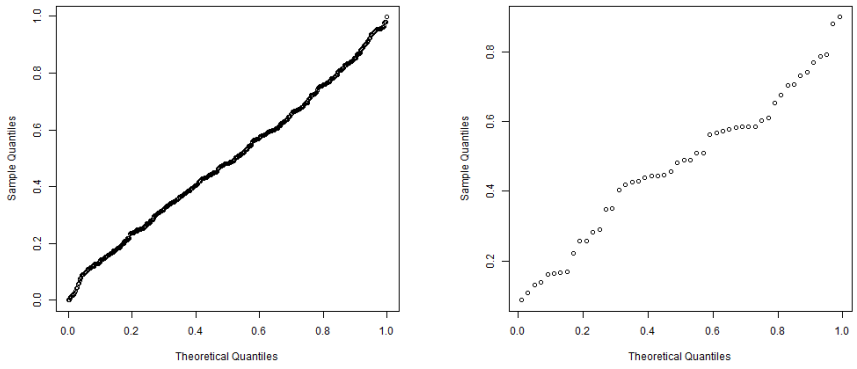
5.3.3 Clustering effects

In Figure 5.9 we can see that the Q-Q plots aiming to reveal any potential serial issues follows a uniform distribution perfectly. That means that the predictor does not suffer from any clustering effects; it quickly adapts to new situations.

FGBM
Mar16



Adidas
AG



Nordex
AG

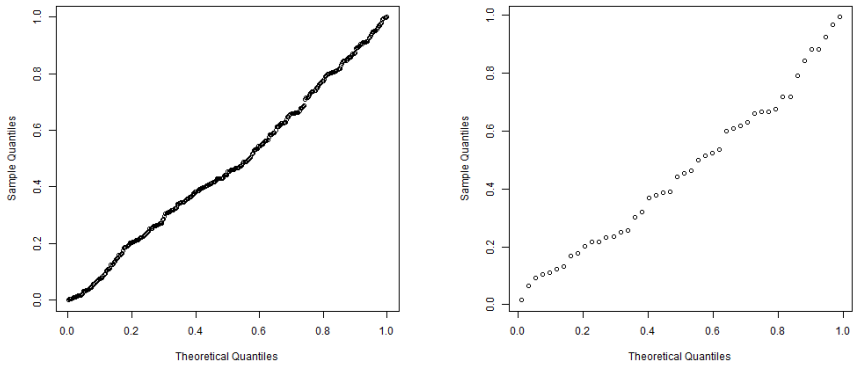


Figure 5.8. Uniform Q-Q plots for the probabilities that the predicted Ψ_t is lower than or equal to the realized ψ_t for three financial instruments. The left plots are taken with sampling length $\Delta t = 1$ min and the right plots are taken with sampling length $\Delta t = 10$ min. There's a strong uniform behaviour in all of the cases, best seen in the predictions for the equities.

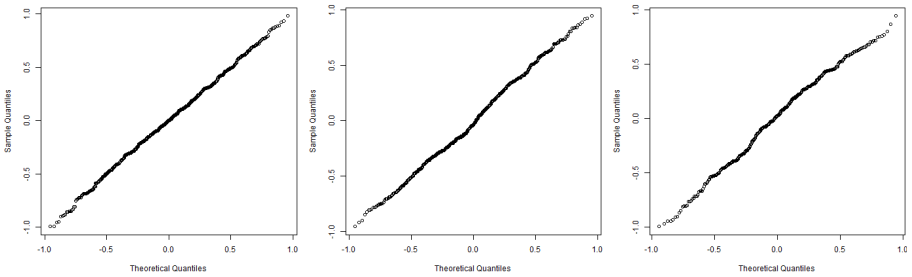


Figure 5.9. Q-Q plots of Δu_t against the theoretical distribution of the difference between two uniformly distributed random variables for FGBM Mar16 (left), Adidas AG (middle) and Nordex AG (right) on Feb 3, 2016, using a sampling interval $\Delta t = 1$ min. In none of the three cases can any clustering effects be identified.

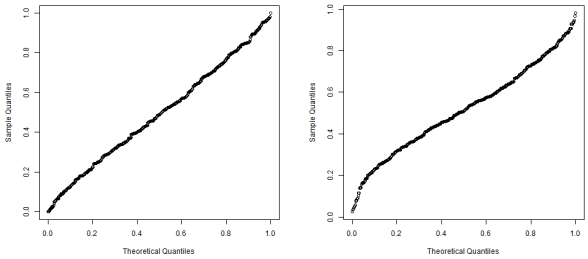


Figure 5.10. Q-Q plots for the probabilities that the predicted Ψ_t is lower than or equal to the realized ψ_t when b are half their optimal values (right) and twice their optimal values (left) for trading in Adidas AG on Feb 3, 2016. The sampling length used is $\Delta t = 1$. The estimate remains good for smaller θ but gets worse as more variance is added.

5.3.4 Hyperparameter sensitivity

As discussed above, the MLE for θ varies across days (see Figure 5.4), which could be problematic with our choice of θ_k^*. In the standard case, we will get good performance with this set-up, as seen above, but we also want to explore the edge-cases. For this reason, we will perform a quick sanity check to see how the predictions are affected by halving or doubling the values of b, compared to their same-day MLE values. We will only consider same-day MLE values of σ here since these estimates are quite stable across days.

In Figure 5.10 we show the distributions of the cumulative probabilities for observed ψ_{t+1} compared to the predicted values. We see that when b is half the optimal value, we still obtain good performance. This might indicate that the transition kernel specified in the model is diffusing a bit faster than necessary. When looking at the plot for b being twice the optimal values, we can see that the variance added by this diffusion results in considerably fatter tails for the predictor.

Chapter 6

Discussion

In this chapter we will discuss the results that we have found, the choices that we have made and what to think about when conducting similar type of research.

6.1 Notes on the framework

The framework is intentionally modular in design. It is perfectly possible to replace the bootstrap filter with a more advanced one—for example, a filter using an adaptive re-sampling scheme. This adds further value to the framework and makes it more future-proof.

We have chosen to implement relatively simple forward and backward filters in this thesis. However, they are both robust and well-studied, which we appreciate. Trying to make everything overly complicated when not necessary is generally a bad thing. Therefore, we stick to simplicity for now, but at the same time keep the door open for improvements.

6.1.1 On-line inference

In this thesis, we have only considered the most simple approximation of θ_k^*. Even with this approximation, we generally obtain a well-performing predictor. The only down-side seem to be that we get unnecessarily fat tails in the predictive distribution at times. This could indicate either that the Laplacian transition kernel is creating excessive noise, or that we simply need to make sure the hyperparameter approximation is not overestimating the b parameters. Apart from the slightly fat tails, the predictor produces consistent results and does not suffer from clustering issues.

6.1.2 Posterior p-values

In this thesis we have focused on the full day's sample of p-values, called $\{u_t\}_{t=1}^T$. This has been used to show that the predictions are consistently good throughout the day. Further, we have shown how to formally test the tail behaviour of the

predictive distribution—with good results. By formulating similar (one-sided) tests, it would be possible to detect anomalies; days with unpredictably high dynamics.

In addition to looking at the full day, each of these values convey something in their own right. Each value u_t represents the cumulative probability for its associated observed value ψ_t. A value close to zero or one means that the observed value is highly unexpected, given the available history of observations. This could be used for finding interesting events that are hard to explain under this model.

6.2 Data handling

When analysing actual trading data it is of utmost importance to handle the data with care. Even if the data is good, which is very rare, there are many pitfalls to be aware of.

In this thesis we have focused purely on analysis of executed trades. However, there are several different kinds of executed trades. Since we are trying to model the intra-day dynamics, we have chosen to only consider trades being executed in the time period between the opening auction and the closing auction, excluding the trades executed during any of those auctions. Some markets also have intra-day auctions that you have to be aware of, as well as pre- and post-trading.

Further, we have also chosen to not consider *over-the-counter* (OTC) trades, as these are not part of the regular central limit order book trading. Depending on the data source, OTC trades might be blended in with the electronic trades, potentially adversely impacting the analysis because of their distinct characteristics (huge volumes, out-of-line prices, etc.).

6.3 Notes on the scaled volume imbalance

The scaled volume imbalance is a very promising quantity. Looking back at Section 2.1.5 we can see that our scaled volume processes Q^B and Q^S are simply aggregations of the trade-by-trade price impacts found by Lillo et al (up to a proportionality constant), having the concavity constant β set to $1/2$. This value both corresponds well to the empirical evidence in the literature, and makes the distribution of the resulting scaled volumes ν express a very clear exponential behaviour.

Since we are looking at a very small scale, using the aggregated trade-by-trade price impacts suggests that Ψ could better represent the information conversion rate, or the information *flow* in the markets, than the standard volume imbalance.

A common practice when analysing the traded volumes, is to not distinguish between buyer-initiated and seller-initiated volumes. We argue that this is not optimal since the incentives for buying and selling are very different, leading to a fundamental asymmetry in these two. For example, in the *FDAX Jun14* data from April 28, 2014 the average (pooled) volumes were 3.1 for selling, but only 1.8 for buying. At the same time, the total seller-initiated volume and the total buyer-

initiated volume, over the whole day, were roughly the same. Since the power-law is not additive this asymmetry gives rise to effects that are hard to fit.

By studying buyer-initiated and seller-initiated trades separately, we have found meaningful and interesting patterns in how they behave, reflecting this asymmetry.

6.4 Intra-day changes

There are several intra-day effects that motivates the use of a dynamic model. We have previously mentioned the diurnal seasonality. Another thing that affects intra-day trading is the steady flow of information in our connected society. By allowing the arrival intensities to vary throughout the day, we can observe what is actually going on in the markets.

This also means that we are able to observe sudden changes in the flow of information. For example when news are released, we can expect the dynamics to be pushed to their boundaries. This should allow for easy extension of the framework that we have developed to allow for detection of news events, such as releases of quarterly reports or M&A's. Any event where the dynamics are violated, meaning that we see sudden unexpected jumps in the latent variables, indicates some type of anomalous event.

When performing our prior sensitivity analysis we limited ourselves to only look at changes in b. We are motivating this choice by the fact that the relative magnitude for the changes in μ is much smaller than for the λ. Also, the σ hyperparameters were all much more stable across different days, than the b hyperparameters.

6.5 Sampling parameters

There are two parameters that we have used in this thesis that are not really part of the set of hyperparameters. Those parameters are the *sample interval length* Δt and *pooling threshold* τ. The reason why we have excluded them from the model is that they concern how the observations are sampled, rather than the model itself. Because of this, we have chosen to call these parameters the *sampling parameters*.

6.5.1 The pooling threshold

The choice to use $\tau = 1$ ms is very simplistic, but still manages to capture the things we need it for. Rather than searching for an optimal values, it would probably be better to extend the method for pooling trades to, for example, identify recurrent fixed-lag arrivals.

6.5.2 The sample interval length

Contrary to τ, the sample interval length could be more interesting to examine closer. In this thesis, we have assumed that all dynamics depend purely on real time. This choice was heavily based on a pre-study that we made on data from

2014. State movement in event time seemed to be much more volatile. Because modelling in real time is also more intuitive, this was then the obvious choice.

As we are looking at high-frequency trading in this thesis, we have set on rather small values for Δt to capture rapid changes in the markets. However, there is no point in having a sampling time that is significantly smaller than the time it takes for the underlying processes to change, since that would only introduce a lot of unnecessary variance. Our results show that the optimal hyperparameters θ grow with Δt. This is intuitive, since a bigger Δt means that we evolve the Markov chain less frequently, potentially resulting in bigger jumps. On the other hand, it increases sub-linearly. This can be explained by a smoothing argument. By observing longer periods of time each time, sporadic instantaneous trading bursts will average out over the whole period.

In this thesis, we do not set out to find a sweet spot for the optimal value of Δt. The choice of Δt might also depend on the specific goals for the intended analysis. For example, the optimal value for Δt when trying to create a stable estimator for θ_{t+1} might not be the same as the optimal for studying goodness-of-fit for alternative transition kernels. It might even be sensible to define an adaptive sampling period that is updated throughout the day, "gluing together" trading days to allow for continuous on-line learning.

6.6 Information carried by trades

In this section we will discuss the connection between the model presented in this thesis and the concept of information in the markets. We have tried to make sure that the quantities used in the model all carry information, and the way they are utilized is designed to extract this information.

We will start by a short reasoning around the importance of trades in the context of risk, as we mentioned in Remark 1. Then we will present discussions on what exact attributes of the trade that actually carries this information and how this is captured in our model.

6.6.1 Actions and risk

Considering the three possible actions listed in Section 6.6, neither (A), nor (B) is necessarily associated with any immediate risk. After placing a limit order it is possible to, at any time prior to execution, cancel the order without any financial loss. Therefore, there are strategies associated with placing and (possibly) cancelling a limit order that is not necessarily tied to information about the particular asset.

On the other hand, by placing a market order the trader will be executing a trade and, hence, be subject to direct financial risk. This pronounced demarcation in associated financial risk for the different actions suggests that the market orders will carry information to a much higher extent than limit orders. We can think of the information carried in trades as *realized information*.

That this action would carry the most information out of the three is also consistent with the reasoning in [3, Section 6.1]. That conclusion might, however, come as a little counter-intuitive, since it is argued that: "the trade carries little new information". To understand this, we must realize that the argument only refers to the fact that trades might not supply any new information *in addition to* what is already present in the limit orders. Nothing is said about the overlap in information between the orders and trades.

6.6.2 Trade direction

The primary information carrying attribute is the direction of the trade; whether it is buyer-initiated or seller-initiated. The direction reveals something very fundamental of the trader's belief in regards to the asset. If you, for example, buy an asset it means that you expect it to not decrease in price. This is captured by the usage of two separate processes Q^B and Q^S as defined in our model. In, for example, [6], direction is the only attribute that is considered.

6.6.3 Trade volume

The other important attribute is the *volume* of the trade. Normally, the more shares you buy, the more certain you are that your belief is correct regarding the direction of future price movements. However, buying 10 shares does not necessarily mean that you have 10 times more information than if you would only have bought 1 share. This is the same as the concavity discussed in Section 2.1.4.

In our model, volume plays a central role, but used under a concave transform, implying that the amount of information is better represented in our model than in, for example, [18].

6.6.4 Trade price

The last interesting attribute is the *price* of the trade. Decoding the information contained carried in the price is however not as straightforward as it might sound. The first piece of information the price conveys is that the trader believes the price to be "favourable" in the sense that by buying at price x, the trader believe the price to not fall below x. On the other hand, this is the same information that is already conveyed by the direction of the trade.

The second piece of information the price conveys is *risk*. This is motivated by the following reasoning: by comparing the traded price to a *real price* the traded price would convey information about the trader's belief in the risk of unfavourable future changes to the price. This is equivalent to information about associated risk and not about the price itself. Since a *real price* cannot be observed it is generally taken to be the mid-price, which is analogous to observing the size of the *spread* at the time of the trade. Thinking of it this way, the connection to risk is more apparent.

In the context of price impact we will therefore argue that it might be sensible to not consider the trade's price as (a) it is difficult to quantify the information correctly and (b) the traded price primarily conveys information about risk, but without a model for the fair price for the asset, it's useless.

Chapter 7

Conclusions and Future work

In this chapter we will go through the major contributions of this thesis and potential openings for future extensions on these results.

7.1 Conclusions

By using hidden Markov models and applying a specialized Monte Carlo machinery including some of the most recent particle-based methods; the bootstrap filter, backward sampling and sequential Monte Carlo expectation-maximization, we have shown that it is possible to build a holistic framework for analysing some of the hottest topics in modern market microstructure theory.

Within this framework we have defined the scaled volume imbalance Ψ. By drawing from previous work in the field of limit order book modelling, along with derivations of clever assumptions, we have developed an interesting HMM for this process, to study using this framework. By utilizing the Monte Carlo machinery, we have successfully produced accurate predictions and shown that our model satisfy the other criteria of interest as well, such as absence of clustering effects and stability in regards to the prior.

Regarding the assumptions that we have made, based on our findings, we can justify the use of the memorylessness property in a dynamic setting. This provides an excellent alternative to the standard approach of using a power-law distribution, which is harder to fit and struggles to describe intra-day features such as the diurnality. In particular, we have found that modelling the trade generation as two inhomogeneous Poisson processes, separating between the buyer-initiated and seller-initiated trades, works very well.

Due to the flexibility of this framework anyone can create and assess similar non-linear hidden Markov models for market behaviour that has only been studied in a static context before. This provides a natural bridge between market microstructure theory (which seeks to describe observed relationships) and limit order book modelling (which tries to model the underlying dynamics of trading).

Any hyperparameters associated with proposed hidden Markov models can be

learnt from data by employing the sophisticated SMCEM algorithm, providing greater insight in their mechanics. Further, based on these models, inference and prediction can be made on-line with very high performance, both in terms of accuracy and with respect to computational cost.

7.2 Future work

Since the angle taken in this thesis is by some means new territory, especially in the field of market microstructure, we believe that this could open up many interesting opportunities for future extensions.

Regarding the framework, the most immediate thing to address would be to investigate how to define a better approximation θ_k^* for the hyperparameter θ_k, given only historic data. Also, studies of how to choose Δt would be very interesting.

For future work in relation to the scaled volume imbalance and its associated model, we would recommend trying to establish the connection to price impact as a starting point. The scaled volume imbalance has been defined to carry the highest degree of market information possibly—from a market microstructure theory point-of-view. If we are lucky, it should have the potential to outdo the standard volume imbalance in this area.

The connection to information suggests that the scaled volume imbalance could be very useful for detecting news events or trading anomalies, just by analysing the trades. A possible way to go around doing that would be to look for sudden violations of the state dynamics of the HMM. This can be motivated by the interpretation of the scaled volume imbalance as conveying the *rate of information flow*. Any unforeseen changes to this flow would then indicate some external event that is not part of the model, for example, a news release or a potentially fraudulent action by any of the traders, resulting in unforeseen trading activity. Another approach, as discussed earlier, would be to adopt a daily hypothesis test to detect days with extreme dynamics.

To improve the performance of the predictions, and to reduce variance, the model could be extended to allow for correlations between the movements of the λ processes. Such correlations have been discussed in literature. Another possible modification would be to compare our HMM with models having alternative choices of transition kernels. For example, what would happen if μ was more or less stationary?

And, finally, it would of course be very interesting to try this framework for other quantities associated with market microstructure, too. As long as appropriate assumptions regarding the underlying dynamics are known, any observable quantity can be studied, making this a novel tool for predicting market phenomena.

Chapter 8

Appendix

8.1 Proofs

8.1.1 EM Algorithm proof outline

The derivation of the EM-algorithm is rather involved but we will outline the steps generally taken to motivate it below. Please note that we will not provide the proof for convexity of the optimization problem.

To start off, take the logarithm of the likelihood function $p_\theta(y_{1:T})$ as it is expressed in Equation 2.5

$$\log\left(p_\theta(y_{1:T})\right) = \log\left(p_\theta(x_{0:T}, y_{0:T})\right) - \log\left(p_\theta(x_{0:T} \mid y_{0:T})\right)$$

Proceeding by multiplying by the density $p_{\theta'}(x_{0:T} \mid y_{0:T})$ on both sides of the equation above and computing the expectation of the unobservable data $x_{0:T}$, with respect to any parameter $\theta' \in \Theta$, we obtain the following expression

$$\log\left(p_\theta(y_{1:T})\right) = \mathcal{Q}(\theta, \theta') - \mathcal{H}(\theta, \theta') \tag{8.1}$$

Here we have introduced the two entities $\mathcal{Q}$ and $\mathcal{H}$, which are known as the *auxiliary quantity* and the *entropy*, respectively. These are defined by

$$\mathcal{Q}(\theta, \theta') = \mathbb{E}_{\theta'}\left[\log\left(p_\theta(x_{0:T}, y_{0:T})\right) \mid y_{0:T}\right]$$

and

$$\mathcal{H}(\theta, \theta') = \mathbb{E}_{\theta'}\left[\log\left(p_\theta(x_{0:T} \mid y_{0:T})\right) \mid y_{0:T}\right]$$

By subtracting the log-likelihood for θ' from Equation 8.1 we obtain

$$\log\left(p_\theta(y_{1:T})\right) - \log\left(p_{\theta'}(y_{1:T})\right) = \left(\mathcal{Q}(\theta, \theta') - \mathcal{Q}(\theta', \theta')\right) + \left(\mathcal{H}(\theta', \theta') - \mathcal{H}(\theta, \theta')\right)$$

In this expression, since θ and θ' both belong to the same space Θ, it is possible to identify $\mathcal{H}(\theta', \theta') - \mathcal{H}(\theta, \theta')$ as the *Kullbeck-Leibler divergence*

$$\mathcal{H}(\theta', \theta') - \mathcal{H}(\theta, \theta') = K(p_{\theta'}(x_{0:T} \mid y_{0:T}) \parallel p_\theta(x_{0:T} \mid y_{0:T})) \tag{8.2}$$

Then, by applying Gibb's inequality the following result is obtained

$$\mathcal{H}(\theta', \theta') - \mathcal{H}(\theta, \theta') \geq 0,$$

which leads to the inequality

$$\log\left(p_\theta(y_{1:T})\right) - \log\left(p_{\theta'}(y_{1:T})\right) \geq \mathcal{Q}(\theta, \theta') - \mathcal{Q}(\theta', \theta'),$$

For the parameter θ^*, which maximizes $\mathcal{Q}(\theta, \theta')$ with respect to θ, the following inequality holds

$$\mathcal{Q}(\theta^*, \theta') \geq \mathcal{Q}(\theta', \theta'),$$

by which we can conclude that the EM-algorithm produces a sequence of parameters $\{\theta_k, k = 1, 2, \ldots\}$ for which the log-likelihood function is non-decreasing.

Bibliography

[1] D. B. Rubin A. P. Dempster, N. M. Laird. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[2] L. Bauwens and P. Giot. *Econometric Modelling of Stock Market Intraday Activity*. Advanced Studies in Theoretical and Applied Econometrics. Springer, 2001.

[3] Jean-Philippe Bouchaud, J. Doyne Farmer, and Fabrizio Lillo. How markets slowly digest changes in supply and demand. In Thorsten Hens and Klaus Schenk-Hoppé, editors, *Handbook of Financial Markets: Dynamics and Evolution*, chapter 2, pages 57 – 160. North-Holland, San Diego, 2009.

[4] Olivier Cappé, Eric Moulines, and Tobias Ryden. *Inference in Hidden Markov Models (Springer Series in Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

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

[6] Rama Cont, Sasha Stoikov, and Rishi Talreja. A stochastic model for order book dynamics. *Operations Research*, 58(3):549–563, 2010.

[7] Randal Douc, Aurélien Garivier, Eric Moulines, and Jimmy Olsson. Sequential Monte Carlo smoothing for general state space hidden Markov models. *The Annals of Applied Probability*, 21(6):2109–2145, 2011.

[8] Arnaud Doucet, Nando De Freitas, and Neil Gordon. *Sequential Monte Carlo methods in practice*. Springer-Verlag, 2001.

[9] Arnaud Doucet and Adam M Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering*, 12(656-704):3, 2009.

[10] J. E. Handschin and D. Q. Mayne. Monte carlo techniques to estimate the conditional expectation in multi-stage non-linear filtering. *International Journal of Control*, 9(5):547–559, 1969.

[11] V.Yu. Korolev, A.V. Chertok, A.Yu. Korchagin, and A.I. Zeifman. Modeling high-frequency order flow imbalance by functional limit theorems for two-sided risk processes. *Applied Mathematics and Computation*, 253:224 – 241, 2015.

[12] Fabrizio Lillo, Stanislaw M. Farmer, J. Doyne, and Rosario N. Mantegna. Econophysics: Master curve for price-impact function. *Nature*, 421(6919):129–130, January 2003.

[13] Fabrizio Lillo, Szabolcs Mike, and J. Doyne Farmer. Theory for long memory in supply and demand. *Phys. Rev. E*, 71:066122, Jun 2005.

[14] Fredrik Lindsten. *Particle filters and Markov chains for learning of dynamical systems*. PhD thesis, Linköping University, Automatic Control, The Institute of Technology, 2013.

[15] Nicholas Metropolis and Stanislaw M. Ulam. The Monte Carlo Method. *Journal of the American Statistical Association*, 44(247):335–341, September 1949.

[16] Maureen O'hara. *Market microstructure theory*, volume 108. Blackwell Cambridge, MA, 1995.

[17] Jimmy Olsson, Olivier Cappé, Randal Douc, and Éric Moulines. Sequential Monte Carlo smoothing with application to parameter estimation in nonlinear state space models. *Bernoulli*, 14(1):155–179, 2008.

[18] Vasiliki Plerou, Parameswaran Gopikrishnan, Xavier Gabaix, and H. Eugene Stanley. Quantifying stock-price response to demand fluctuations. *Phys. Rev. E*, 66:027104, Aug 2002.

[19] Donald B. Rubin. Estimation in parallel randomized experiments. *Journal of Educational Statistics*, 6(4):377–401, 1981.

[20] Donald B. Rubin. Bayesianly justifiable and relevant frequency calculations for the applies statistician. *The Annals of Statistics*, 12(4):1151–1172, 1984.

TRITA -MAT-E 2017:29
ISRN -KTH/MAT/E--17/29--SE